Application of Convolution Neural Networks in Web Search Log Mining for Effective Web Document Clustering

Suruchi Chawla, Shaheed Rajguru College Delhi University, India*

ABSTRACT

The volume of web search data stored in search engine logs is increasing and has become big search log data. The web search log has been the source of data for mining based on web document clustering techniques to improve the efficiency and effectiveness of information retrieval. In this paper, deep learning model convolution neural network (CNN) is used in big web search log data mining to learn the semantic representation of a document. These semantic documents vectors are clustered using K-means to group relevant documents for effective web document clustering. Experiment was done on the data set of web search query and associated clicked URLs to measure the quality of clusters based on document semantic representation using deep learning model CNN. The clusters analysis was performed based on WCSS (the sum of squared distances of documents samples to their closest cluster center) and decrease in the WCSS in comparison to TF. IDF keyword-based clusters confirm the effectiveness of CNN in web search log mining for effective web document clustering.

KEYWORDS

Convolution Neural Network, Information Retrieval, K-Means Clustering, Web Documents, Web Search Engines

1. INTRODUCTION

The volume of web data is increasing rapidly every day and is responsible for the information overload problem. (Gantz & Reinsel ,2012)The artificial intelligence techniques have been applied to big data to obtain the abstract representation of the knowledge present in data for various applications. (Adomavicius & Tuzhilin, 2005)

Documents clustering techniques are used for improving the efficiency and effectiveness of Information retrieval. Use of partition document clustering for information retrieval improves the retrieval efficiency as the document collections are partitioned and queries are matched against cluster centroids only. The retrieval efficiency is achieved by reducing the number of query-document comparisons for IR, but there is decrease in retrieval effectiveness. Retrieval effectiveness is the percentage of relevant documents retrieved (Salton, & Buckley,1988) . Hybrid of optimization techniques like ACO as well as trust, Genetic Algorithm and Ontology have been used for effective personalized web search. (Chawla ,2016 ; Chawla, 2018)

Deep learning models are widely used in big data mining to identify the abstract semantic feature from low level input data. The input data vector is passed through successive layers of non linear

DOI: 10.4018/IJIRR.300367 *Corresponding Author

transformation to generate the high level semantic abstraction. These semantic representations of web documents and queries are used as effective source of knowledge for fast and effective information retrieval. Deep learning technique like convolution neural network has been used effectively to extract the semantic representation of web search queries and clicked documents. CNN proves to be effective in learning of semantic and patterns from queries, documents, users and items. (Shen et al., 2014)

In (Xu, He & Li, 2018) convolution neural network is used to learn document as well as query semantic feature vector of low dimensionality for search as well as neural collaborative filtering models for recommendation. K-means has been simple and efficient for wide variety of data types. K-means has low computational requirements and store only documents, cluster membership of the documents and the cluster centroids. (DeFreitas, & Bernard, 2015)

In this paper deep learning model convolution neural network(CNN) is used in web query session mining to generate the abstract document semantic vector. The resulting semantic vectors are further clustered using K-means clustering to reveal search patterns of web users and is evaluated for clusters quality.

Experiment is conducted on the data set of web search query sessions for analyzing the effectiveness of deep learning model convolution neural network on the quality of cluster of web documents. The results of cluster analysis based on WCSS has been compared with TF.DF based clusters. The results show that WCSS decreases drastically for clustering using CNN based document representation therefore confirms the improvement in clusters quality using CNN based document semantic representation.

The organization of paper is as follows section 2 provides a detailed survey of related work, section 3 covers basic concepts used in the paper, section 4 provides the details of proposed work, section 5 explains the experimental study and in section 6 conclusion of paper is described.

2. RELATED WORK

Deep Learning algorithms had been used to model the abstract representation of data from non-linear and non-trivial user-item relationship present in available abundant data sources. In document retrieval and classification systems, TF.IDF and BM25 were used to represent the document based on word count where each word was considered to be independent however individual words were highly correlated for inferring the document semantics. (Cheng et al.,2016;Song, Elkahky, & He, 2016)

Deep learning technique CNN had been used to generate the abstract semantic representation of document using convolution -max pooling followed by non linear transformation through successive hidden layers . The resulting document semantic vector is of reduced dimensionality and thus improve the computational efficiency as well as effectiveness of documents retrieval. Deep learning techniques had been widely applied in domains like information retrieval, web search, natural language processing, question answering, image retrieval and speech recognition. (Bansal, Belanger, & McCallum, 2016; Peng et al.,2017 ; LeCun , Bengio , & Hinton, 2015; Li & Lu , 2016; Wu, DuBois, Zheng, & Ester, 2016)

In (Salakhutdinov, & Hinton, 2009) semantic hashing was used to find documents similar to query document. The documents represented as short binary codes were used as memory address. During information retrieval the similar documents selected for retrieval were at memory address with small hamming distance from memory address of query document.

In (Ranzato, & Szummer, 2008) deep learning parameters were learnt based on both Labelled and Unlabelled data. Google Word2Vec tool had been used for the extraction of semantic representation from big data. It used large training input text corpus and generate the word vector as output. In (Mikolov, Chen, Corrado, & Dean, 2013) neural network was used for generation of high quality semantic word vectors based on training. The training was done using huge datasets of distinct words in the vocabulary.

In (Mikolov, Le, & Sutskever, 2013 ; Okura et al., 2017) deep learning-based recommender systems were used for generating user models based on user preferences requirements. The performance

as well as user satisfaction was enhanced. In (Huang et al., 2013 ;Elkahky, Song, & He, 2015; Xu et al., 2016) recommender system based on Deep structured semantic model was used for information retrieval and generates top-n recommendations. Deep Structured Semantic Model(DSSM) had been used for mapping the input vector to semantic vector of low dimensionality and semantic similarity was computed based on cosine function.

In (Pang et al.,2017) deep learning architecture was used for relevance based ranking in information retrieval. In (Soliman, El-Sayed, & Hassan, 2015) an approach was proposed for clustering of search engine results using documents semantics vector based on ontology . The quality of clusters was measured using the precision measure that represents the percentage of documents correctly classified in clusters to the number of documents. Humans conducted manual clustering of search engine results and was used as the reference for evaluating the semantic clustering of search engine results using ontology. The experiment results shows an average precision of 90% across clusters of web documents for different queries.

In (Motta et al.,2019) Computational Model based on CNN was used to identify adult mosquitoes based on the mosquito images features .In (Alaskar et al.,2019) due to popularity of Convolution neural networks in the analysis and identification of medical images. *C*onvolutional neural networks was used for automated ulcer detection.. In (Gu,2019) due to often use of CNN in machine learning area for its strong and efficient machine learning ability as well as automatic feature extraction. Thus CNN model had been applied for security code recognition. In (Dashdorj & Song,2019*)* deep learning algorithms learnt to get optimal combination of feature set and hyperparameter setting for classification. In (Ashesh,Pedram & Saba,2020) an effective auto-labeling strategy was proposed for measuring the performance of CNN in predicting the clusters . The results confirmed the effectiveness of CNN in predicting the clusters. The training of deep CNN shows an accuracy for both identification and prediction. The performance of CNNs was also evaluated take into consideration of impact of architecture and its hyperparameters.

3. BACKGROUND

3.1 Architecture of Convolution Neural Network for the Generation Of Abstract Representation Of Document Semantic Vector

Deep learning Convolution neural network model has been widely used in extracting the semantic representation of text using sequence of complex non linear transformation applied to input data.

The document content is represented as matrix based on concatenation of its Word2Vec word embedding. The clicked documents of web query sessions are processed and represented as a matrix where each row represents a pre-trained Word2Vec vectors for each distinct word of vocabulary present in document.

Word embedding mapped the words of document to vector of real numbers of low dimensionality based on Natural Language Processing (NLP). Word2Vec has been used as the commonly used Word embedding techniques and identifies the context of a word in a document . (Mikolov, Chen, Corrado, & Dean, 2013;Pennington, Socher, & Manning, 2014)

In CNN convolution operation is applied that involves applying the filter $W_c \in R^{h1k}$ to window of h1 words in input X having total T words where each word $x_i \in R^k$ of size k and a new feature map is produced as follows.

$$C1 = f\left(W_c \times X + b\right) \in R^{T-h1+1} \tag{1}$$

where f is the non linear hyperbolic tangent function and bias vector $b \in R^{T-h1+1}$

A given filter is applied to all words in input matrix by sliding the filter of height h1 words based on stride size and produced a feature map. The application of multiple filters on input matrix generate multiple feature maps and complementary document features are captured at various level of abstraction.

Thus these multiple feature maps are the output of the convolution operation. The output of the convolution layer is a variable length feature map that depends on the size of the input word sequence. The activation function such as tanh introduced nonlinearities that is required for multi-layer networks. (Kim, 2014)

The max-pooling operation (Collobert et al., 2011) is applied to a feature map and extract a maximum value, i.e., $\hat{c}1 = \text{maximum}\{c1\}$, as the feature .Thus for each feature map, pooling captured the most important feature removing less informative compositions of words and the extracted features are fixed length. The function of a pooling layer generates the fixed length feature vector by combining maximum value from each feature map and generate the abstract representation of all feature maps produced from convolution layer. The fixed length feature vector generated using max- pooling layer is fed to fully connected softmax layer that gives the probability distribution over labels in the output layer.

3.2 Training of CNN Based on Clickthrough Web Query Session Data For The Generation Of Semantic Document Concept Vector

The data set of clickthrough data is partitioned to validation and training set. During training of CNN , convolution matrix W_c, FNN weights as well as bias were adjusted based on backpropagation through stochastic gradient descent for maximizing the probablity of relevant document given search query. That is, to reduce the following loss function value .

$$L\left(\Lambda\right) = -log\prod_{\left(Q,D^{+}\right)} P\left(D^{+} \mid Q\right) \tag{2}$$

$$P\left(D^{+} \mid Q\right) = \frac{exp\left(\gamma R\left(Q,D^{+}\right)\right)}{\sum_{D'\epsilon\ D} exp\left(\gamma R\left(Q,D'\right)\right)} \tag{3}$$

$$R\left(Q,D\right) = cosine\left(y_Q, y_D\right) = \frac{y_Q^T y_D}{\| y_Q \| \| y_D \|} \tag{4}$$

where $D1^{+}$ is the set of clicked web document and D1 is obtained using $D1^{+}$ and $J1$ is set of random selection of unclicked documents $\{\text{-}\ D_j^{-}; j=1..J1\}$

y_Q and y_D are the concept vector of query and document. The $R\left(Q,D\right)$ is a relevance score computed using cosine similarity between query Q and document D vectors. ^ denotes the parameter set of Convolution latent semantic model(CLSM) to learn that include W_c (convolution matrix) and FNN weights as well as bias. Thus, the output layer of fully connected layer maps the query/document to semantic vector in the lower dimensional vector space.

The region size of filter is varied in range 3 to 5 and it was observed that increasing the region size beyond 3 will not improve the performance and do not provide extra context. It is found that

with little hyperparameter tuning of CNN on pretrained word vector shows the significant results on multiple benchmarks.(Mitra, 2015)

The number of filters are varied in the range of 100 to 600. It is recommended that increasing the number of filters in this range will increase the running time and using filters beyond 600 is not suggested. It is found that if the best value lies on the border of the range then large values should be used.(Bengio, 2012)

3.3 K-means Clustering of CNN Based Documents Semantic Vector

K-means clustering algorithm is capable of handling large data sets and is simple to use . K-means algorithm has been effectively used for clustering document vectors. (Rai & Singh,2010; DeFreitas, & Bernard, 2015; Xu & Tian, 2015)

Algorithm:
K-means: the K-means algorithm is partitional method of unsupervised clustering. The cluster's centres are computed based on the average value of the data vectors in a given cluster.

Input Data: K: the prediction of number of clusters D2: data set of n vectors
Output: A set of K clusters and their centroids:

1.Select randomly K vectors from D2 as the initial cluster's centers;
2. Repeat for every x vector in D2
3. Find the cluster whose centroid vector is most similar to the x and then label x with the selected cluster.
4. Update the cluster centroids, that is, average of the data objects vector belonging to each cluster;
5. Until there is no change in clusters centroids

4. PROPOSED METHOD

In this paper method has been proposed using convolution neural network model in clickthrough web search query log mining and the document semantic vectors are extracted for effective clustering. These document semantic vectors improves the quality of clustering therefore increases the number of relevant documents in clusters for effective web information retrieval. The CNN uses convolution layer on clicked document matrix in order to capture the complementary as well as multiple features of clicked documents at various level of abstraction. CNN has been widely used for feature extraction from massive amount of data based on non linear model and the semantics of data is extracted which otherwise is not possible with linear models. Multiple filters of different heights are used to capture the local features at various level of abstraction followed by maxpooling operation. Max-pooling operation retain global feature by selecting the highest value from each feature map. This global feature vector is fed to non-linear FNN and the semantic vector of reduced dimensionality is generated. These document concept vectors are clustered to improve the quality of clustering for effective information retrieval. Thus improvement in the quality of clusters therefore increases the number of relevant documents in clusters for effective web information retrieval.

The data set of web search query sessions consists of search query and its clicked URLs. It is assumed that clicked document is atleast partially relevant to search query. The query/document word vector is mapped to matrix using Word2Vec embedding. The document matrix is input to convolution layer which applies multiple filters of a given region size and capture features of documents at various level of abstraction. These resulting multiple feature map of varying length is used by maxpooling operation to generate the fixed length vector. This fixed length vector is input to non linear hidden layer in FNN and generate the semantic vector at the output layer.

The optimization of semantic concept vector is done at the output layer based on maximizing the likelihood of relevant document given the user search query. The K-means clustering of these

document concept vector generated using CNN improve the quality of clusters. Since the paper is focused on analyzing the cluster quality based on document content representation due to this reason comparison is done using content representation TF.IDF versus CNN based semantic vector. Since the clustering algorithm is not the focus of comparison therefore K-means algorithm is used for clustering because it is simple and efficient for wide variety of data types. K-means has been used because of low computational requirements and has low memory requirements. These clusters of document concept vectors group relevant clicked documents in a given domain for effective Information Retrieval.

The Stepwise Execution of the Proposed Method Is Given Below

1. Data set of click through log containing search query and clicked URLs is preprocessed where each clicked document/query is represented using TF.IDF representation.
2. The bag of word representation of document is mapped to matrix using Word2Vec model.
3. The document matrix is applied as input to CNN convolution layer that uses multiple filters of a given region size to capture the document features at various level of abstraction.
4. For every application of filter, the variable length feature map is generated.
5. Maxpooling operation extracts maximum feature value from each feature map and the global abstract fixed length feature vector is generated .
6. This fixed length feature vector is fed to fully connected neural network and undergoes non linear transformation to generate semantic vector.
7. The semantic concept vector of documents is optimized during training of CNN using backppropagation based on maximizing the likelihood of relevant document given search query.
8. The resulting documents concept vectors are clustered to group documents based on semantics.
9. The cluster analysis based on WCSS is done to access the quality of clusters of web documents for effective web information retrieval.

5. EXPERIMENT

The data set of web search query sessions containing search query and clicked URLs was collected through the GUI interface developed as shown in Fig 1. The user search query issued for web search to GUI interface retrieves the search results shown with check boxes. The user's response given through check boxes were stored as search query session. The clickthrough data containing search query and its clicked URLs was preprocessed using TF.IDF vector model. The document keyword vector was represented as word matrix using Word2Vec embedding for each distinct word present in the document.

The data set was collected for the duration of 3 months from November 2019 to January 2020 for training the CNN models for the extraction of document concept vector. There were 15,392 queries session collected for training the model and 10,000 queries sessions were collected as validation data set for optimization of hyperparameters of trained model. Thus on an average search query is associated with 25 clicked URLs.

During training of CNN model, the convolution matrix W_c , weights as well as bias were learnt using backpropagation stochastic gradient descent based on maximizing the probability of relevant document given the search query. The trained model was run on validation data set for the optimization of hyper-parameters such as number of filters, filter region size ,activation function and maxpooling strategy. The snapshot execution of CNN in tensorflow for generation of document concept vector is shown in Fig 2.

The optimal results of model were obtained for following values of hyperparameters filter region size =3, number of filters =300, tanh activation function and maxpooling for extracting global feature. The learning rate of FNN was set to 0.0001. The clicked URLs were mapped to abstract semantic document vector using trained CNN. Table 1 shows the CNN parameters and optimal values selected during training.

Figure 1. Shows the GUI interface where the user's clicks to search results are captured using checkboxes.

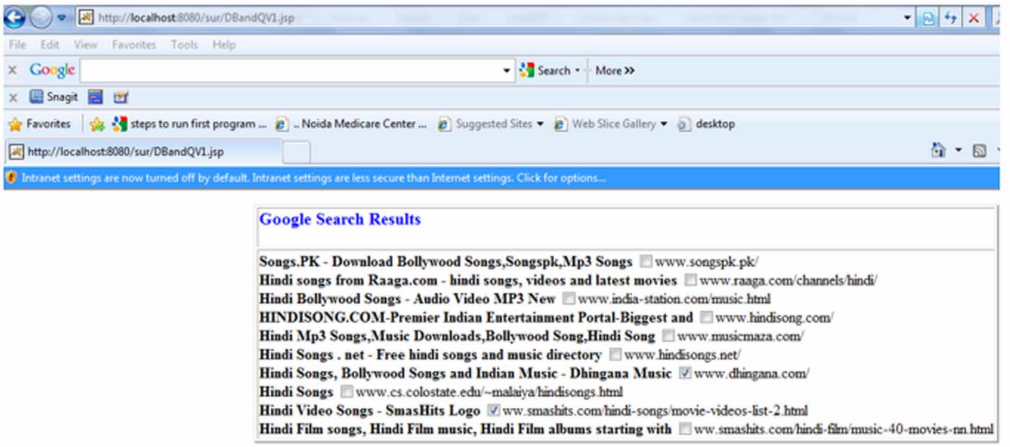
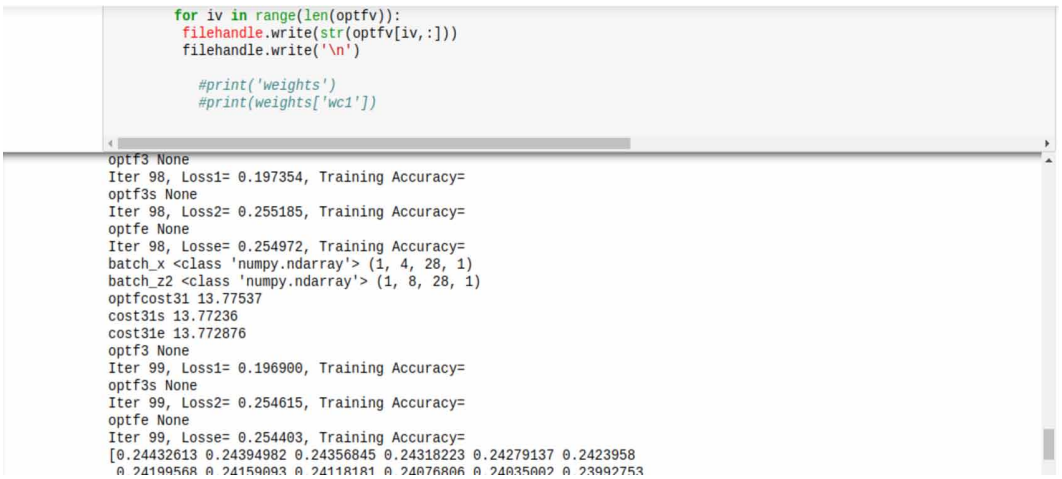


Figure 2. Shows the execution of CNN in tensorflow for the generation of document concept vector.



K-means clustering was used for clustering clicked document concept vector generated using CNN/TF.IDF. WCSS is good measure of cluster quality based on sum of squared distances of samples to their closest cluster center as given in eq(5).

Table 1 CNN configuration showing the parameters and their optimal values selected during training.

Parameter	Values
Filter size(h1)	{1,3,5,7} Optimal value =3
Number of filters	{100,200,300} Optimal value=300
Activation function	Tanh
Pooling	Maxpooling
Dimension of convolution and maxpooling layer	300
Stride size	1
Number of input neuron in FNN	300
Number of hidden layer	1
Number of neurons in output layer of FNN	128
Learning parameters	Convolution matrix W_c and weight matrix W_s (weights and biases) of fully connected layer
FNN learning method	Backpropagation through stochastic gradient descent.

$$WCSS = \sum_{k=1}^{K} \sum_{i1 \in C_k} \sum_{j1=1}^{p} \left(v_{i1j1} - \overline{v_{kj1}} \right)^2 \tag{5}$$

K is the number of clusters , i1 is the index of data point belonging to cluster C_k ($k \in 1..K$ p is the length of each data point vector $v_{i1j1} j1 \in 1..p$) assigned to clusters. $\overline{v_{kj1}}$ ($j1 \in 1..p$) is the centroid vector of cluster C_k Low value of WCSS for a particular value of K implies low dissimilarity of data points within clusters therefore indicates high quality of clusters in comparison to clusters having high value of WCSS.

WCSS is the output of K-Means clustering class of sklearn.cluster module in python. WCSS was computed in reference to clusters of document vectors using (CNN/TF.IDF) obtained using K-Means. Table 2 shows the WCSS values for different values of K for both TF.IDF/CNN based clusters of web documents. Fig 3 as well as Fig 4. display the graph showing the WCSS (measure of clusters quality) for different value of K using CNN based semantic / Simple TF.IDF based web document clusters. The x shows the number of clusters K that varies from 1 to 4. The y axis shows WCSS that is measure of dissimilarity of data points from within clusters to their respective cluster centers for different value of x.

Fig 4. shows that for CNN based document semantic vector, the WCSS decreases more rapidly in comparison to TF.DF document vector. The low value of WCSS signifies that within cluster dissimilarity of data points to their cluster centers decreases therefore it confirms that CNN extract the web document vector close to its semantics therefore effective document clustering using CNN based document semantic vector representation. Using elbow method (use elbow of the curve to choose the optimal number of clusters) K=3 is selected in CNN based Clustering .Thus CNN proves to be effective in generating the semantic representation of document vectors in comparison to TF.IDF vector and hence improves the quality of clusters. These clusters can be further used for information retrieval during web search.

Table 2 Shows the WCSS values for different values of K for both TF.IDF/CNN based clusters of web documents.

Number clusters (K)	Web content (TF.IDF/CNN) based Document Clustering	
	WCSS Values(sum of squared distances of samples to their closest cluster center)	
	TF.IDF based Simple Clustering	CNN based Clustering
K=1	6.43	2.66e-05
K=2	4.61	1.87e-05
K=3	3.41	1.16e-05
K=4	2.33	7.64e-06

Figure 3 Shows the clusters quality measured using WCSS versus number of clusters using TF.IDF based Simple Clustering.

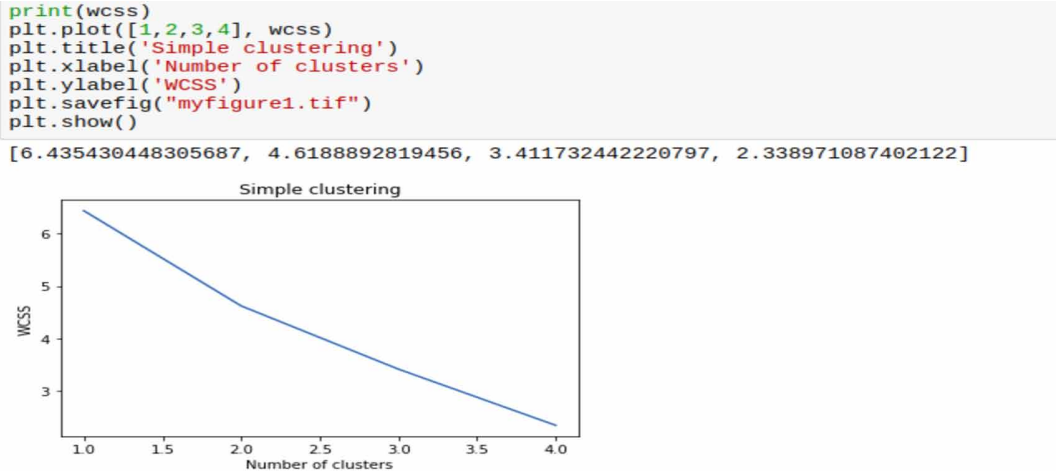
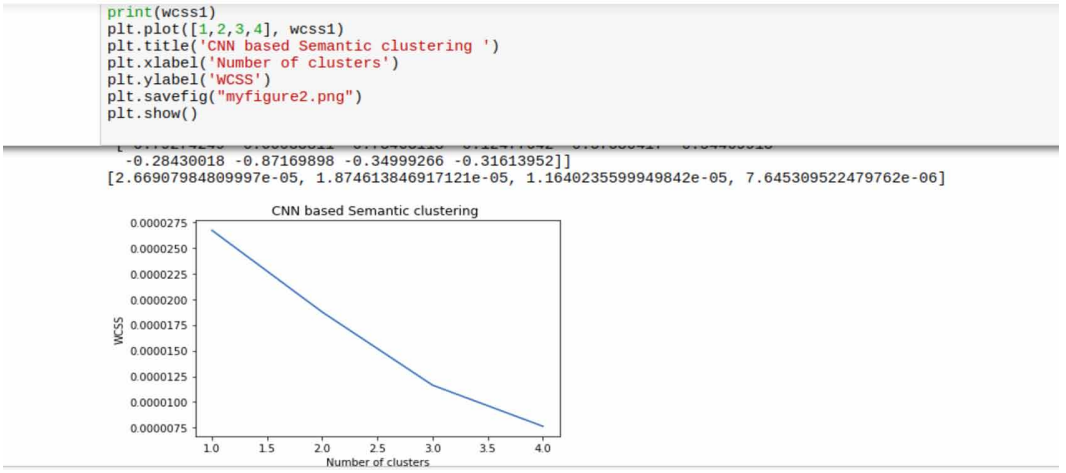


Figure 4. Shows the clusters quality measured using WCSS versus number of clusters using CNN based Semantic clustering.



The decrease in WCSS value is more in CNN based clustering than TF.IDF based web document clustering this shows the improvement in clusters quality through the use of CNN in web user search data mining and hence more and more relevant documents are grouped in clusters for effective information retrieval. Thus CNN extract the document semantic vector at abstract level, thus these compact document representations learnt are efficient as they require fewer computations and also less storage capacity. The quality of clusters was measured using the precision. Precision is the measure of percentage of documents correctly classified in clusters to the number of documents. The average precision of clusters based on TF.IDF/CNN based document representation was computed based on the average precision of recommended search results using clusters of web documents with 25 search queries in selected domain using eq 6,7,8 and also shown in Table 3. Thus the cluster quality was accessed was based on users clicks to recommendation of clicked documents of clusters using GUI interface as shown in Fig 1. The user clicks to search results measures the fraction of cluster documents that are relevant and correctly classified to cluster . The average precision of all clusters based on recommended web documents (TF.IDF/CNN based clusters) using eq 9 access the overall quality of clusters.

Assuming majority of users does not search beyond the first web page containing 10 search results therefore average precision @ 10 is computed that measure the precision of recommended 10 search results thus giving weightage to top ranked relevant documents.

$$Precision = \frac{number\ of\ relevant\ documents}{number\ of\ retrieved\ documents} \tag{6}$$

$$Precision\ of\ cluster_i = \frac{number\ ofr\ elevant\ documents\ present \in cluster_i}{total\ number\ of\ documents\ present \in cluster}$$

$$\frac{number\ of\ clicked\ recommended\ documents\ of\ cluster_i}{total\ number\ of\ recommeded\ documents\ of\ cluster_i} \tag{7}$$

$$Average\ precision\ of\ cluster_i = \frac{\sum_{t=Q1}^{Q25} Precision\ of\ cluster_i}{25} \tag{8}$$

$$Average\ precision\ of\ Clustering\ solution\left(K=3\right) = \frac{\sum_{i=1}^{3} Average\ precision\ of\ cluster_i}{3} \tag{9}$$

The average precision of clusters of web documents semantic vector based on CNN was computed as shown in Table 3 based on eq(6)(7)(8)(9) and found to be 93% hence shows improvement in cluster quality in comparison to shown in (Soliman, El-Sayed, & Hassan, 2015) for semantic clustering based on ontology Thus high value of average precision of clusters based on recommended search results using CNN based clusters confirms that CNN effectively capture the semantic of web documents using abstract representation with reduced dimensionality therefore improving the effectiveness of clustering algorithm in grouping relevant documents for information retrieval.

Table 3 Shows the average precision of recommended search results using TF.IDF/CNN based clusters of web documents in three domains Academics, Sports and Entertainment.

Search queries used for Information Retrieval based on clusters	**Average Precision@10 based on recommended search results using TF.IDF /CNN based clusters of web documents.(K=3)**					
	TF.IDF Simple Clustering			CNN based Semantic Clustering		
	Academics	Sports	Entertainment	Academics	Sports	Entertainment
Q1	0.29	0.49	0.63	0.94	0.96	0.97
Q2	0.49	0.73	0.74	0.89	0.94	0.94
Q3	0.63	0.73	0.73	0.89	0.89	0.96
Q4	0.74	0.63	0.74	0.94	0.94	0.93
Q5	0.74	0.74	0.73	0.89	0.96	0.94
Q6	0.73	0.74	0.63	0.89	0.94	0.96
Q7	0.63	0.49	0.63	0.94	0.96	0.94
Q8	0.62	0.74	0.74	0.89	0.94	0.94
Q9	0.48	0.63	0.49	0.94	0.94	0.88
Q10	0.49	0.74	0.64	0.89	0.88	0.89
Q11	0.29	0.29	0.49	0.94	0.96	0.96
Q12	0.49	0.74	0.73	0.89	0.94	0.96
Q13	0.49	0.74	0.64	0.94	0.94	0.94
Q14	0.74	0.73	0.74	0.89	0.94	0.89
Q15	0.74	0.74	0.64	0.97	0.89	0.94
Q16	0.63	0.63	0.49	0.89	0.94	0.94
Q17	0.29	0.64	0.49	0.97	0.94	0.89
Q18	0.29	0.64	0.64	0.89	0.94	0.97
Q19	0.74	0.73	0.74	0.94	0.88	0.94
Q20	0.49	0.74	0.49	0.89	0.88	0.96
Q21	0.48	0.73	0.64	0.97	0.94	0.94
Q22	0.62	0.64	0.74	0.89	0.89	0.96
Q23	0.73	0.74	0.64	0.97	0.94	0.94
Q24	0.63	0.49	0.49	0.89	0.94	0.96
Q25	0.64	0.74	0.74	0.97	0.94	0.96
Average precision	0.56	0.65	0.64	0.92	0.93	0.94
Average precision of clusters	0.62			0.93		

6. CONCLUSION

In this paper CNN is applied in web query session log mining for generating the abstract semantic representation of document vectors for clustering relevant documents for effective web information retrieval. The document matrix representation was input to convolution as well as maxpooling layer and the fixed length feature vector was generated. This fixed length feature vector was passed to fully connected layer which undergoes non linear transformation at various layer and the high level semantic document vector was generated. These document semantic vector were clustered using K-means algorithm for grouping relevant documents in clusters. Experiment was performed to assess the quality of clusters using CNN based web document concept vectors. The decrease in WCSS value confirms the success of CNN in capturing the semantics of document /query vector at abstract level using context of words in document/query vector in comparison to manually design features(TF.IDF) for clustering relevant documents. Furthermore the results of the average precision of clusters of web documents based on recommendation of web documents of clusters confirms the high precision of recommended search results using CNN based web document clusters. Therefore CNN based web document clustering groups relevant web pages together that can be further used for effective information retrieval.

REFERENCES

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, *17*(6), 734–749. doi:10.1109/TKDE.2005.99

Alaskar, H., Hussain, A., Al-Aseem, N., Liatsis, P., & Al-Jumeily, D. (2019). Application of convolutional neural networks for automated ulcer detection in wireless capsule endoscopy images. *Sensors*, *19*(6), 1265, 1-16.

Ashesh, C., Pedram, H., & Saba, P. (2020). Predicting clustered weather patterns: A test case for applications of convolutional neural networks to spatio-temporal climate data. *Scientific Reports*, *10*(1). PMID:31992743

Bansal, T., Belanger, D., & McCallum, A. (2016, September). Ask the gru: Multi-task learning for deep text recommendations. *Proceedings of the 10th ACM Conference on Recommender Systems*, 107-114. doi:10.1145/2959100.2959180

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade* (pp. 437–478). Springer. doi:10.1007/978-3-642-35289-8_26

Chawla, S. (2016). A novel approach of cluster based optimal ranking of clicked URLs using genetic algorithm for effective personalized web search. *Applied Soft Computing*, *46*, 90–103. doi:10.1016/j.asoc.2016.04.042

Chawla, S. (2018). Ontology-Based Semantic Learning of Genetic-Algorithm Optimised Back Propagation Artificial Neural Network for personalized web search. *International Journal of Applied Research on Information Technology and Computing*, *9*(1), 21–38. doi:10.5958/0975-8089.2018.00003.9

Cheng, H. T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., & Anil, R. (2016, September). Wide & deep learning for recommender systems. *Proceedings of the 1st workshop on deep learning for recommender systems,* 7-10. doi:10.1145/2988450.2988454

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, *12*(Aug), 2493–2537.

Dashdorj, Z., & Song, M. (2019). An application of convolutional neural networks with salient features for relation classification. *BMC Bioinformatics*, *20*(10), 244.

DeFreitas, K., & Bernard, M. (2015). Comparative performance analysis of clustering techniques in educational data mining. *IADIS International Journal on Computer Science & Information Systems*, *10*(2), 65–78.

Elkahky, A. M., Song, Y., & He, X. (2015, May). A multi-view deep learning approach for cross domain user modeling in recommendation systems. *Proceedings of the 24th International Conference on World Wide Web,* 278-288. doi:10.1145/2736277.2741667

Gantz, J., & Reinsel, D. (2012). The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. *IDC iView: IDC Analyze the future*, *2007*(2012), 1-16.

Gu, J. (2019, April). The Application of Convolutional Neural Network in Security Code Recognition. In Journal of Physics: Conference Series (Vol. 1187, No. 4, p. 042064). IOP Publishing. doi:10.1088/1742-6596/1187/4/042064

Huang, P. S., He, X., Gao, J., Deng, L., Acero, A., & Heck, L. (2013, October). Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management* (pp. 2333-2338). doi:10.1145/2505515.2505665

Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882. doi:10.3115/v1/D14-1181

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. doi:10.1038/nature14539 PMID:26017442

Li, H., & Lu, Z. (2016, July). Deep learning for information retrieval. *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval,* 1203-1206.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space.* arXiv preprint arXiv:1301.3781.

Mikolov, T., Le, Q. V., & Sutskever, I. (2013). *Exploiting similarities among languages for machine translation.* arXiv preprint arXiv:1309.4168.

Mitra, B. (2015, August). Exploring session context using distributed representations of queries and reformulations. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval* (pp. 3-12). doi:10.1145/2766462.2767702

Motta, D., Santos, A. Á. B., Winkler, I., Machado, B. A. S., Pereira, D. A. D. I., Cavalcanti, A. M., Fonseca, E. O. L., Kirchner, F., & Badaro, R. (2019). Application of convolutional neural networks for classification of adult mosquitoes in the field. *PLoS One*, *14*(1), e0210829. doi:10.1371/journal.pone.0210829 PMID:30640961

Okura, S., Tagami, Y., Ono, S., & Tajima, A. (2017, August). Embedding-based news recommendation for millions of users. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* 1933-1942. doi:10.1145/3097983.3098108

Pang, L., Lan, Y., Guo, J., Xu, J., Xu, J., & Cheng, X. (2017, November). Deeprank: A new deep architecture for relevance ranking in information retrieval. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management,* 257-266. doi:10.1145/3132847.3132914

Peng, Y. X., Zhu, W. W., Zhao, Y., Xu, C. S., Huang, Q. M., Lu, H. Q., Zheng, Q., Huang, T., & Gao, W. (2017). Cross-media analysis and reasoning: Advances and directions. *Frontiers of Information Technology & Electronic Engineering*, *18*(1), 44–57. doi:10.1631/FITEE.1601787

Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 1532-1543. doi:10.3115/v1/D14-1162

Rai, P., & Singh, S. (2010). A survey of clustering techniques. *International Journal of Computers and Applications*, *7*(12), 1–5. doi:10.5120/1326-1808

Ranzato, M. A., & Szummer, M. (2008, July). Semi-supervised learning of compact document representations with deep networks. *Proceedings of the 25th international conference on Machine learning*, 792-799. doi:10.1145/1390156.1390256

Salakhutdinov, R., & Hinton, G. (2009). Semantic hashing. *International Journal of Approximate Reasoning*, *50*(7), 969–978. doi:10.1016/j.ijar.2008.11.006

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, *24*(5), 513–523. doi:10.1016/0306-4573(88)90021-0

Shen, Y., He, X., Gao, J., Deng, L., & Mesnil, G. (2014, April). Learning semantic representations using convolutional neural networks for web search. *Proceedings of the 23rd International Conference on World Wide Web,* 373-374. doi:10.1145/2567948.2577348

Soliman, S. S., El-Sayed, M. F., & Hassan, Y. F. (2015). Semantic clustering of search engine results. *The Scientific World Journal*. PMID:26933673

Song, Y., Elkahky, A. M., & He, X. (2016, July). Multi-rate deep learning for temporal recommendation. *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval,* 909-912.

Wu, Y., DuBois, C., Zheng, A. X., & Ester, M. (2016, February). Collaborative denoising auto-encoders for top-n recommender systems. *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining,* 153-162. doi:10.1145/2835776.2835837

Xu, D., & Tian, Y. (2015). A comprehensive survey of clustering algorithms. *Annals of Data Science*, *2*(2), 165–193. doi:10.1007/s40745-015-0040-1

Xu, J., He, X., & Li, H. (2018, June). Deep learning for matching in search and recommendation. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval,* 1365-1368. doi:10.1145/3209978.3210181

Xu, Z., Chen, C., Lukasiewicz, T., Miao, Y., & Meng, X. (2016, October). Tag-aware personalized recommendation using a deep-semantic similarity model with negative sampling. *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 1921-1924. doi:10.1145/2983323.2983874