Enhanced Approach for Information Retrieval

Sanket Bhave
Computer Science
Colorado State University
Fort Collins, USA
sanket.bhave@colostate.edu

Shalini Durga Royyuru
Computer Science
Colorado State University
Fort Collins, USA
shalini2@colostate.edu

Sina Mahdipour Saravani
Computer Science
Colorado State Univeristy
Fort Collins, USA
Sina.Mahdipour_Saravani@colostate.edu

Indrakshi Ray
Computer Science
Colorado State University
Fort Collins, USA
Indrakshi.Ray@colostate.edu

***Abstract*—Information and Document Retrieval has witnessed significant advancements in the last few decades. Starting from statistical methods like Term Frequency-Inverse Document Frequency (TF-IDF) to developing convolutional neural networks for word embedding, we have seen vast changes in the way data can be looked upon. In this paper we have reviewed various approaches to the information retrieval and compared them. In the end we have proposed a methodology which combines the advantages of two different deep learning models i.e., Bi-directional Encoder Representations from Transformers (BERT) and Duet Model. We also propose to work on real-world data for information retrieval. We are confident that proposed methodology will gain accuracy then the existing ones.**

Keywords—Information Retrieval, BERT, TF-IDF, Duet Model

I. INTRODUCTION

Due to vast use of Internet in our daily lives we see a significant boost in data creation and data usage. Text data forms significant portion of whole data generated. Major search engines like google handle about 1.2 trillion searches every year. These search engines have made it easy to access information for users. Text search engines harness the technology of information retrieval for document querying. Information Retrieval (IR) is software program responsible for storage, organization, optimization, and retrieval of document on a particular query. The IR algorithm selects and ranks the documents according to the statistical calculations for each query. Statistical similarity in a ranked query is used to identify the closeness of each document to the query.

Previously various indexing techniques and statistical analysis methods were used for information retrieval. The concept of Term Frequency- Inverse Document Frequency (TF-IDF) was majorly used as a part of statistical analysis and indexing. Later, recent advancements in natural language processing and deep learning explored new ways to understand complex patterns in the language. This led to more efficient retrieval system. The introduction of pre trained models like Bi-directional Encoder Representations from Transformers (BERT) and XLNet demonstrate a novel method of pre training language representations

In this paper, we will look at different methods tried for information retrieval and their comparisons. Also, we will include the results of our experiment. At the last we will propose new methods on which we plan to work in future.

II. LITERATURE SURVEY

The need to store and retrieve text information has become very vital over the past years. Now a days important information like research papers, company reports, survey reports etc. have become available on a single click.

Many researchers have worked and are still working to improve the efficiency of these IR systems. In 2006, Justin Zobel and Alistair Moffat [4] described the use of TF-IDF for document retrieval. To understand TF-IDF we need to knowledge about some relevant statistical terms. Most similarity measures use a small number of fundamental statistical values:

- $f_{d,t}$, the frequency of term t in document d;
- $f_{q,t}$, the frequency of term t in the query;
- f_t, the number of documents containing one or more occurrences of term t.
- F_t, the number of occurrences of term t in the collection.
- N, the number of documents in the collection; and
- n, the number of indexed terms in the collection.

TF-IDF does three tasks:

(1) Less weight is given to terms that appear in many documents.

(2) More weight is given to terms that appear many times in a document; and

(3) Less weight is given to documents that contain many terms.

Following values are calculated through the query and text corpus:

$$w_{q,t} = \ln\big(1 + (N \div f_t)\big) \qquad w_{d,t} = 1 + \ln f_{d,t}$$

Here, $W_{q,t}$ is called inverse document frequency and $W_{d,t}$ captures the term frequency.

TF-IDF is product of term frequency (TF) and inverse document frequency (IDF)

$$TF - IDF = TF \times IDF$$

TF-IDF is calculated for each term in the corpus. After that cosine similarity or any similarity measure can be applied to get the most relevant documents from the corpus. In the mentioned paper the authors use TF-IDF to design an indexing algorithm. Here the authors have proposed the use of data structures *Inverted File:* a collection of lists, one per term, recording the identifiers of the documents containing that term.

´Further, the concept of Baseline Inverted File is defined and used for text-based search. A baseline inverted file index consists of two major components. The search structure or vocabulary stores for each distinct word t,

- a count f_t of the documents containing t, and
- a pointer to the start of the corresponding inverted list.

The second component of the index is a set of inverted lists where each list stores for the corresponding word t,

- the identifiers d of documents containing t, represented as ordinal document numbers; and
- the associated set of frequencies $f_{d,t}$ of terms t in document d.

By using TF-IDF and the above data structure the author was able to reduce the retrieval time of the traditional IR systems.

The concept of TF-IDF is even implemented now for document retrieval which is evident through [4].

Further as the concept of machine learning and data mining came into picture, they were implemented for information retrieval as Siham JABRI, et.al [9], Qing Liu et.al [7]. In both papers authors use traditional data mining techniques like Apriori algorithm and clustering to find the features from the underlying text corpus to find patterns within the data and match the patterns with user query.

As we can see there was a shift from traditional statistical methods to applying data mining techniques for information retrieval system.

The introduction of neural networks has further improved the efficiency of the IR systems. Moreover, the study of natural language processing which can represent complex sentence structure in numerical form has enhanced the way we looked towards text retrieval. [5], [6], [8], [11]. In [6] Po-Sen Huang et.al have proposed a deep neural network architecture for mapping the raw text features into a semantic space. The authors here have used the concept word hashing to reduce the dimensionality of the bag-of-words term vectors. By using the concept n-grams the authors here have constructed feature vectors. By training DNN with these feature vectors we get their corresponding semantic concept vectors.

Finally, by applying cosine similarity among query vectors and document vectors the authors have retrieved the top-K documents. This was a phenomenal improvement over TD-IDF modelling as the improvements over accuracy for TF-IDF was more than 4%. Many papers hence forth have utilized same concept of constructing feature vector as introduced in the above paper.

The paper by Yelong Shen et.al [10] have demonstrated the use of convolutional neural networks for feature extractions and document ranking. Here instead of extracting features separately and passing them to neural network model the author harnesses the use of convolutional neural network (CNN) for feature extraction. This was a huge change in the perspective in which we look towards the data. Many authors used CNN for feature extraction and document ranking.

Later, Bhaskar Mitra, Fernando Diaz, and Nick Craswell proposed a new model based on the studies from [3] they call this new model as the duet model. This duet model architecture combines the strength of two different models which the author refers as local model and distributed model. The local model by its name helps in extracting local features of the query and the corpus while the distributed model learns lower dimensional vector representation of the same. This method gained huge popularity due its simplicity accuracy and effectiveness. In this methodology the local model extracts the exact match and position in the query document. While the distributed model uses n-gram as is used in [3]. Further these features are passed into CNN model which was originally proposed by [10]. This architecture improved the performance over baseline models by more than 4%.

In the local model we tried to find the exact match of every query term in the document. This gives us the matrix with indicates the position of the term with respect to the index in the document. This interaction matrix is passed through a convolutional layer, and then through two fully connected layers, a drop-out layer and final fully connected layer that produces a single real valued output.

In the distributed model the duet uses a character n-graph base representation of each term in the query as well as in the document as proposed in [3]. For both the query and the document the method performs convolution. After convolution these is a max pooling and then a fully connected layer. To perform the matching element wise or Hadamard product between the embedded document matrix and the query is used. After this the matrix is passed through fully connected layers and a drop-out layer until a single real value score is obtained.

Here after many methodologies like [12] have used CNN and n-grams for feature extraction.

Introduction of pre-trained models like BERT have brought in phenomenal invention on the field of word embedding and feature extraction. Many researchers like [11] have enhanced the power of BERT to improvise their existing document ranking system. Many people have also experimented BERT on standard datasets like MS Marco, TREC-Car [5]. The main innovation of BERT is to apply bidirectional training of transformers to language modelling. Usually, text sequence is looked either in one direction i.e., left to right or right to left. But BERT model has proved that training a language model in bidirectional improves language context and flow than in single direction model.

BERT uses transformers for learning contextual relation between the words. Transformer encoder reads the text sequence in bidirectional way, and this allows the model to understand context of a word based on its surrounding words (left – to – right and right- to – left).

Although deep learning has improved the performance of traditional IR systems, the importance of TF-IDF cannot be underestimated. As visible in the methodology proposed in [8], BERT along with TF-IDF gave a significant improvement in the performance in the IR system. When compared to BERT alone TF-IDF was performing 40 % more accurate; while TF-IDF combined with BERT outperformed than BERT and TF-IDF individually.

Comparison of different deep neural networks models is given in Table 1:

S.No	**Model**	**Performance (NDCG Score)**
1.	L-WHDNN [3]	0.498
2.	CD- SSN [10]	0.447

3.	Duet-Model (unweighted)	0.664
4.	Duet-Model (weighted)	0.53
5.	Conv-KNRM [12]	0.481
6.	BERT + TF-IDF	0.75

TABLE 1. Comparison different deep neural network models

NDCG Score:

Normalized Discounted Cummilative Gain (NDCG) is a measure of ranking quality used in information retrieval. NDCG measures the usefulness of a document based on its position in the result list. It is a normalized version of discounted cummilative gain (DCG).

Example: If given a list of documents in response to a search query a user is asked to rank a document on scale of 3-0 with 3 being not revelant and 3 being highly relevant. For 3 documents d_1, d_2, d_3 if the user provides scores as 2,0,3; following steps are taken to provide NDCG score:

$$Cumulative\ Gain = \sum 2 + 0 + 3 = 5$$

After calculating cumulative gain we should calculate distributed cumulative gain (DCG) .

Doc No	CG	$Log_2(CG + 1)$	DCG for each doc
1	2	1	2
2	0	1.585	0
3	3	2	1.5

TABLE 2. DCG Score Calculation

$$DCG = \sum 2 + 0 + 1.5 = 3.5$$

For calculating the ideal ordering of the above scores in the decreasing order is 3,2,0. The DCG for this order is 4.262.

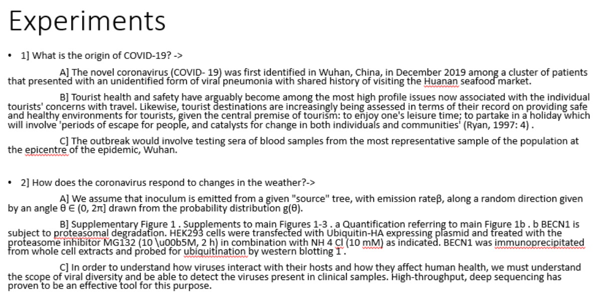
The NDCG of the above ranking will be

$$3.5 / 4.262 = 0.821$$

In this way NDCG score is calculated and the performace of the IR system is measured.

III. Experiments

We performed some experiments on the Duet Model [1] on some sample data and on TREC-CAR dataset. We analyzed the performance of the Duet model and understood how it ranks documents. Given a training set of possible queries and their relevant documents and corresponding irrelevant documents we can train duet on any available datasets. After training if we input some queries, the duet model will give us the similarity score of queries with every document in the corpus. This result and study are an important milestone in the design of our proposed methodology. Although, our work is in progress, we did some experiments and here are the intermediate results:



Experiments

- 1] What is the origin of COVID-19? ->

A] The novel coronavirus (COVID- 19) was first identified in Wuhan, China, in December 2019 among a cluster of patients that presented with an unidentified form of viral pneumonia with shared history of visiting the Huanan seafood market.

B] Tourist health and safety have arguably become among the most high profile issues now associated with the individual tourists' concerns with travel. Likewise, tourist destinations are increasingly being assessed in terms of their record on providing safe and healthy environments for tourists, given the central premise of tourism: to enjoy one's leisure time; to partake in a holiday which will involve 'periods of escape for people, and catalysts for change in both individuals and communities' (Ryan, 1997: 4) .

C] The outbreak would involve testing sera of blood samples from the most representative sample of the population at the epicentre of the epidemic, Wuhan.

- 2] How does the coronavirus respond to changes in the weather?->

A] We assume that inoculum is emitted from a given "source" tree, with emission rateβ, along a random direction given by an angle θ ∈ (0, 2π] drawn from the probability distribution g(θ).

B] Supplementary Figure 1 . Supplements to main Figures 1-3 . a Quantification referring to main Figure 1b . b BECN1 is subject to proteasomal degradation. HEK293 cells were transfected with Ubiquitin-HA expressing plasmid and treated with the proteasome inhibitor MG132 (10 \u00b5M, 2 h) in combination with NH 4 Cl (10 mM) as indicated. BECN1 was immunoprecipitated from whole cell extracts and probed for ubiquitination by western blotting 1 .

C] In order to understand how viruses interact with their hosts and how they affect human health, we must understand the scope of viral diversity and be able to detect the viruses present in clinical samples. High-throughput, deep sequencing has proven to be an effective tool for this purpose.

In future, we plan to work more in this direction.

IV. Proposed Methodology

As visible from the above table the duet model and BERT + TF-IDF model outperform any other known models. In our proposed model we plan to work on combining the duet and BERT architecture. By filtering the results from duet model through BERT we proposed that we get an improved accuracy for our document retrieval system.

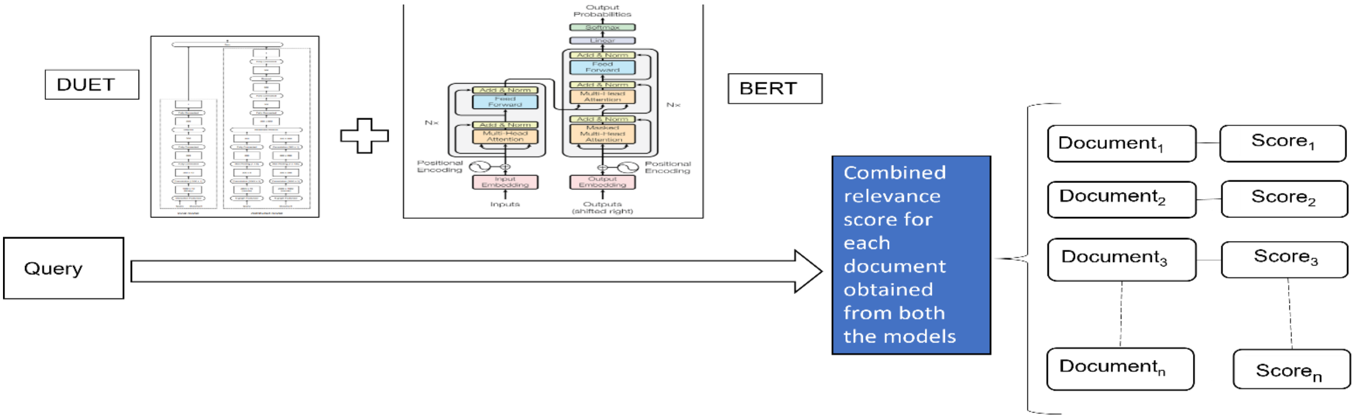


Fig 1. Proposed Methodology

Moreover, if we work on a real-world data like COVID dataset, we can be successful in implementing our combined architecture and can gain a higher accuracy than the existing models.

As is known the duet model combines the local and distributed models of the text corpus and hence achieves higher accuracy then the individual model. BERT is a pre trained model on large Wikipedia corpus. But, from the results given by [8] we see that BERT alone does not perform well on Information retrieval systems.

Hence if we combine the power of both the architectures, we can get a significant improvement in the performance of the new proposed model.

We plan to undertake the following steps for our model:

1. Preprocessing: We processed the data to convert it to lower case and remove some insignificant stop words from it. Also, we planned to use some stemming and lemmatization for incomplete words and to reduce the training size.
2. Feature Vector: We planned to extract the exact position of each and every term in the corpus.
3. Training the Model: We plan to use the feature vectors to train the duet and BERT model.
4. Combining the Scores: We use the score given by the Duet Model as an output as performance major for Duet model and cosine similarity for the BERT model. We will give some weight to both the scores based on how well the individual model performs. The combined score will give the relevant documents to the respective user query.

V. Conclusion

We conclude that the proposed methodology will work better than the existing models. We analyzed that the BERT is not that accurate and combing it with any model will gain more accuracy from referred methodologies. In future we plan to implement the proposed architecture and present the results accurately and in precise form.

References

[1] Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to Match using Local and Distributed Representations of Text for Web Search. In Proceedings of the 26th International Conference on World Wide Web (WWW '17). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1291–1299.

DOI:https://doi.org/10.1145/3038912.3052579

[2] Deng, L., He, X., and Gao, J., 2013. "Deep stacking networks for information retrieval." In ICASSP

[3] Gupta, Y. , Saini, A. , Saxena, A. (2013). 'A Review on Important Aspects of Information Retrieval'. World Academy of Science, Engineering and Technology, Open Science Index 84, International Journal of Computer and Information Engineering, 7(12), 1638 - 1646.

[4] H. Lin, T. Lo and B. Chen, "Enhanced Bert-Based Ranking Models for Spoken Document Retrieval," 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), 2019, pp. 601-606, doi: 10.1109/ASRU46091.2019.9003890.

[5] https://github.com/bmitra-msft/NDRM

[6] https://www.kaggle.com/c/trec-covid-information-retrieval/data

[7] Justin Zobel and Alistair Moffat. 2006. Inverted files for text search engines. ACM Comput. Surv. 38, 2 (2006), 6–es. DOI:https://doi.org/10.1145/1132956.1132959

[8] Nogueira, Rodrigo, and Kyunghyun Cho. "Passage Re-ranking with BERT." arXiv preprint arXiv:1901.04085 (2019).

[9] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In <i>Proceedings of the 22nd ACM international conference on Information &amp; Knowledge Management .Association for Computing Machinery, New York, NY,USA,2333–2338. DOI:https://doi.org/10.1145/2505515.2505665.

[10] Q. Liu, J. Wang, D. Zhang, Y. Yang and N. Wang, "Text Features Extraction based on TF-IDF Associating Semantic," 2018 IEEE 4th International Conference on Computer and Communications (ICCC), 2018, pp. 2338-2343,

Doi: 10.1109/CompComm.2018.8780663.

[11] S. Choudhary, H. Guttikonda, D. R. Chowdhury and G. P. Learmonth, "Document Retrieval Using Deep Learning," 2020 Systems and Information Engineering Design Symposium (SIEDS), 2020, pp. 1-6, doi: 10.1109/SIEDS49339.2020.9106632.

[12] S. Jabri, A. Dahbi, T. Gadi and A. Bassir, "Ranking of text documents using TF-IDF weighting and association rules mining," 2018 4th International Conference on Optimization and Applications (ICOA), 2018, pp. 1-6,

Doi: 10.1109/ICOA.2018.8370597

[13] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. Learning semantic representations using convolutional neural networks for web search. In Proceedings of the 23rd International Conference on World Wide Web (WWW '14 Companion). Association for Computing Machinery, New York, NY,USA, 373–374.

DOI:https://doi.org/10.1145/2567948.2577348.

[14] Yilmaz, Zeynep Akkalyoncu et al. “Applying BERT to Document Retrieval with Birch.” EMNLP/IJCNLP (2019).

[15] Zhuyun Dai, Chenyan Xiong, Jamie Callan, and Zhiyuan Liu. 2018. Convolutional Neural Networks for Soft-Matching N-Grams in Ad-hoc Search. In Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM '18). Association for Computing Machinery, New York, NY, USA, 126–134.

DOI:https://doi.org/10.1145/3159652.3159659