Generic Search Optimizer and Library Books Recommendation System

Chandan Suri[1], Gaurav Sinha[2], Arpita Singh[3], and Dr. Shalini Batra[4]

[1-4]*Computer Science and Engineering Department, Thapar Institute of Engineering and Technology, Patiala*

***

Abstract— Currently, using Thapar's default searching technique, is a frustrating experience for users. The current search implementation on Search Optimization (NNCL – Nava Nalanda Central Library) generates course results by substring matching a user's query against course codes, titles, author names, ISBN, and then displays the results. While not terrible, this strategy often leads to situations where the most relevant books are not on the results page, which is the source of many headaches and labor. Most significantly, the most relevant results are often not even on the first page! Students must search for several different variations of a single concept to find the best books. Additionally, since the current system works by exact substring match, it can be sensitive to slight differences in search terms.

The need to design this model is to improve the experience of users. In this model, the user would enter the query and would get the effective results displayed using the developed model. All the data used will be stored in Central Repositories (server) and will also be updated on the local machine on a day to day basis to train the new datasets.

We chose to pursue this problem because improving search for NNCL will benefit not just ourselves as students but also the whole of the Thapar community. The books we take are integral to our intellectual vitality, so students must have the right tools in hand to find books.

The input to our algorithm is a string representing a user's query (e.g. 'Computer networks'). Using a composite model that we have built, we then generate an ordered list of courses with high relevance to the given search string (e.g. [Networks, ...]).

Keywords— **Word2Vec model, DDC Classification, Clustering based on Subject Tags, Minimum Edit distance, Segmentation, Natural language processing, Neural network, Search optimization, recommender for libraries, books recommender, book ranking, machine learning.**

1. INTRODUCTION

These days recommendation systems are the heart of any eCommerce website, these types of systems help to build such a relationship with their clients, as something that was not possible before. These systems help to get a person to know what he/she wants, even before that individual can think of it, though eventually, he would! Thus, these systems are everywhere today but in some fields, there is still a lot of work to be done.

There are over 70,000 libraries in India till date. Wherever we go, it's the same room with a variety of books, the bounds of which cannot be comprehended. We as humans can't comprehend that, but for machines that would not be the same case! Using techniques of machine learning, we can understand what category a book may fall in but to create a recommendation system for a library is not an easy task, because the categories are unknown, the genres can be known but not the categories. Thus, for each different libraries having books of different domains, another recommendation system is needed. It is developed from scratch repeatedly. There ought to be a different way to this.

In some libraries in India, the search techniques are so poor, they only use the substring matching technique for finding books, and the results are thus, not so good if you are at the library searching for a domain. We have tried to solve the problem to quite an extent by building a generalized recommendation system which would work with any library around the globe once trained on the database of the books there, it would not depend on the domain of the books there, it would work for any kind of database. It would solve the problem related to the searching and the recommendations of the books at different libraries around the globe. Till now, it has a limitation that the database should be having book names in English only, but it can easily be extended to other languages as well.

Though we have currently improved the searching technique of Nava Nalanda Central Library at Thapar University, the composite model created for the implementation applies to any library having books of any domain, as the dataset of Thapar University's library was so rich and diverse in the domain, the training done on the database gave us reliance in the thought of generalizing the recommender for any library around the globe as the input to our algorithm is a simple string representing a user's query.

2. RELATED WORK

The usage of vectors in the field of representation of the meaning of words for analysis has been studied for a long time and quite extensively [1]- [3]. The current state-of-the-art model, Word2Vec is described in [2]. Also, an extension to Word2Vec to obtain vectors for sentences and paragraphs was proposed in [4]. This is done using each paragraph or treating each label as a context word used as an additional input feature. We have used the Kaggle Bag of Model as a part of the composite model created for this project.

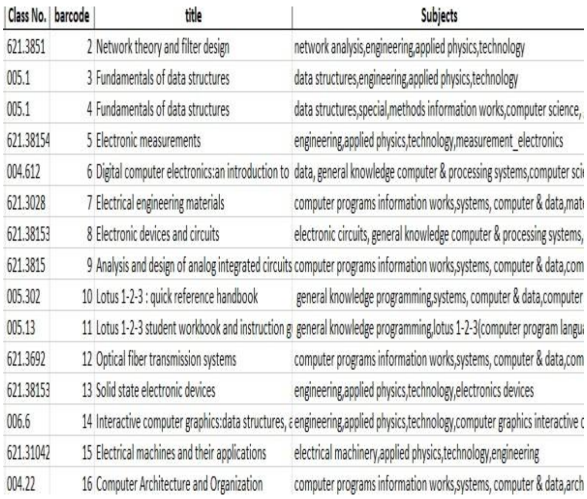
No prior literature exists on improving search especially for NNCL at Thapar University, but in this field of search optimization, some prior work has been done for improving the search in Explore Courses [5]. Usage of word vectors in search engines has also been described [6] and is widespread today. Additionally, neural networks have been applied to “topic spotting” [7], a similar problem to this one. No DDC Classification has been used in this field before and has not been combined with the neural network structure; neither the clustering of the data and finding topics of books has been done before having such a diverse variety of books. Some of the algorithms used for the implementation of the project is the traditional algorithms of natural language processing. The ranking of the results is also done in a novel way as not has been done before using the results of the Word2Vec [8] probabilities.

3. DATASET AND FEATURES

The dataset was provided by NNCL, Thapar University constituting over lakh tuples, each tuple representing a book at Thapar University. The dataset consists of many features, but the relevancy is of only 3 of them. Firstly, we have the field of “Accession Number” which is unique for each book, then comes the field of “Book Title”, and then there is the field of “Subject Tags”, and lastly the field of “DDC”. DDC [9] stands for Dewey Decimal Classification, this is an international standard for all the libraries around the globe, in this each digit stands for a domain of books and thus, we generated more subject tags with this number and thus, used these subject tags for classification.

As you can see in figure 1, the DDC number has 3 digits before the decimal and then digits after the decimal as well, the domains are shown in the figure which is used to generate the subject tags which are then used for classification as well as clustering. Figure 2 shows the subject tags which were related to each digit of the DDC number, it is an international standard, so, this is a standard table around the globe.

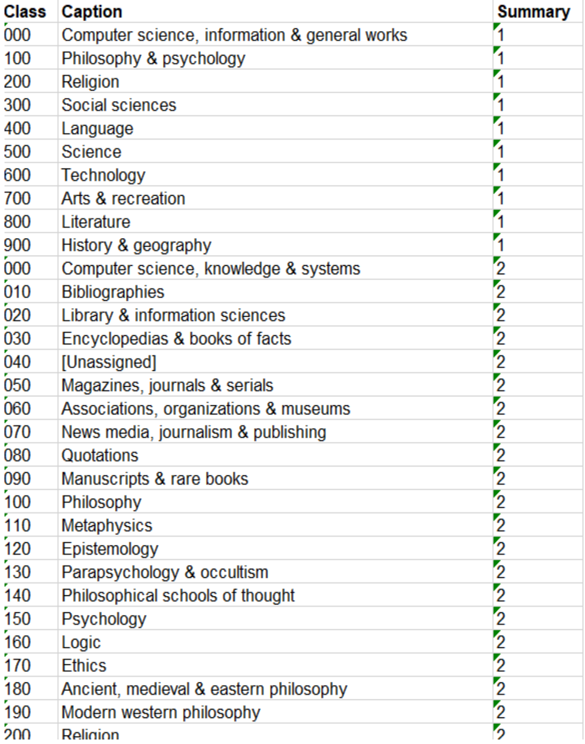
Thus, the book titles present are fed into the Neural network and is also used for correction of a query while DDC is used for classification, along with the Subject tags, clustering is implemented, and Barcode numbers are used to uniquely identify the books and it quite helps in the ranking as well.



Class No.	barcode	title	Subjects
621.3851	2	Network theory and filter design	network analysis,engineering,applied physics,technology
005.1	3	Fundamentals of data structures	data structures,engineering,applied physics,technology
005.1	4	Fundamentals of data structures	data structures,special,methods information works,computer science,
621.38154	5	Electronic measurements	engineering,applied physics,technology,measurement_electronics
004.612	6	Digital computer electronics:an introduction to	data, general knowledge computer & processing systems,computer sci
621.3028	7	Electrical engineering materials	computer programs information works,systems, computer & data,mat
621.38153	8	Electronic devices and circuits	electronic circuits, general knowledge computer & processing systems,
621.3815	9	Analysis and design of analog integrated circuits	computer programs information works,systems, computer & data,com
005.302	10	Lotus 1-2-3 : quick reference handbook	general knowledge programming,systems, computer & data,computer
005.13	11	Lotus 1-2-3 student workbook and instruction g	general knowledge programming,lotus 1-2-3(computer program langu
621.3692	12	Optical fiber transmission systems	computer programs information works,systems, computer & data,com
621.38153	13	Solid state electronic devices	engineering,applied physics,technology,electronics devices
006.6	14	Interactive computer graphics:data structures, a	engineering,applied physics,technology,computer graphics interactive c
621.31042	15	Electrical machines and their applications	electrical machinery,applied physics,technology,engineering
004.22	16	Computer Architecture and Organization	computer programs information works,systems, computer & data,arch

Figure 1. In this figure, you can see the DDC number which is represented by the column name “Class No.” and “barcode” column represents the unique “Accession Number” for each book, then comes the “title” field as well as the “Subject Tags” generated as well as those that were present before, both the tags have been amalgamated into one column for easy referencing for classification.

User may enter a query which would consist of some word which should be present in the title field of any of the books, and even if those words are not present in the title of any books, the also the results are generated effectively for that query, by the neural net used in the composite model. When a book is found final referencing is done using the “barcode” or the accession number thus, giving the results to the user.



Class	Caption	Summary
000	Computer science, information & general works	1
100	Philosophy & psychology	1
200	Religion	1
300	Social sciences	1
400	Language	1
500	Science	1
600	Technology	1
700	Arts & recreation	1
800	Literature	1
900	History & geography	1
000	Computer science, knowledge & systems	2
010	Bibliographies	2
020	Library & information sciences	2
030	Encyclopedias & books of facts	2
040	[Unassigned]	2
050	Magazines, journals & serials	2
060	Associations, organizations & museums	2
070	News media, journalism & publishing	2
080	Quotations	2
090	Manuscripts & rare books	2
100	Philosophy	2
110	Metaphysics	2
120	Epistemology	2
130	Parapsychology & occultism	2
140	Philosophical schools of thought	2
150	Psychology	2
160	Logic	2
170	Ethics	2
180	Ancient, medieval & eastern philosophy	2
190	Modern western philosophy	2
200	Religion	2

Figure 2. This shows the Classes of the books categorized accordingly by the digits present in the class number, each number had some Caption associated with it which summarizes the domain of that field, which is then used for classification and clustering.

4. SYSTEM DESIGN

There were mainly four phases in this project as you can see in figure 3.

A. *Phase 1*

It comprises of the correction of the query using Minimum Edit Distance and Segmentation.

- Developing a spell Checker using Minimum Edit Distance.
- Segmentation of concatenated word based on Probability (using Bigram and Trigram Model)
- Processing the individual words using DDC summarization
- Finding similar books based on the tags generated by DDC summarization of the entered query.
- Generating results based on DDC summarization.

B. *Phase 2*

- Implementing the Neural Network model for finding similar words. (Word2Vec Model)
- Generating results based on Neural Network model
- Developing the low-level weightage function for aggregating Results.
- Aggregating Results based on DDC summarization and Neural Network model.

C. *Phase 3*

It is one of the most important and crucial phases of the project and has a lot of important technical development mainly-Applying Classification technique for clustering of Data based on tags.

- Deciding the training and testing data and applying K fold classification.
- Filtering the results by taking aggregate results of DDC summarization results and Kaggle Bag of Model results.
- Developing the high-level weightage function. This includes the ranking of the results as well.
- Generating results based on clusters obtained.

D. *Phase 4*

- The intersection of the results obtained from phase 1 and phase 3.
- Integrating present SQL model to the newly build Machine Learning Model.
- Generating Aggregate Results based on the user query type.

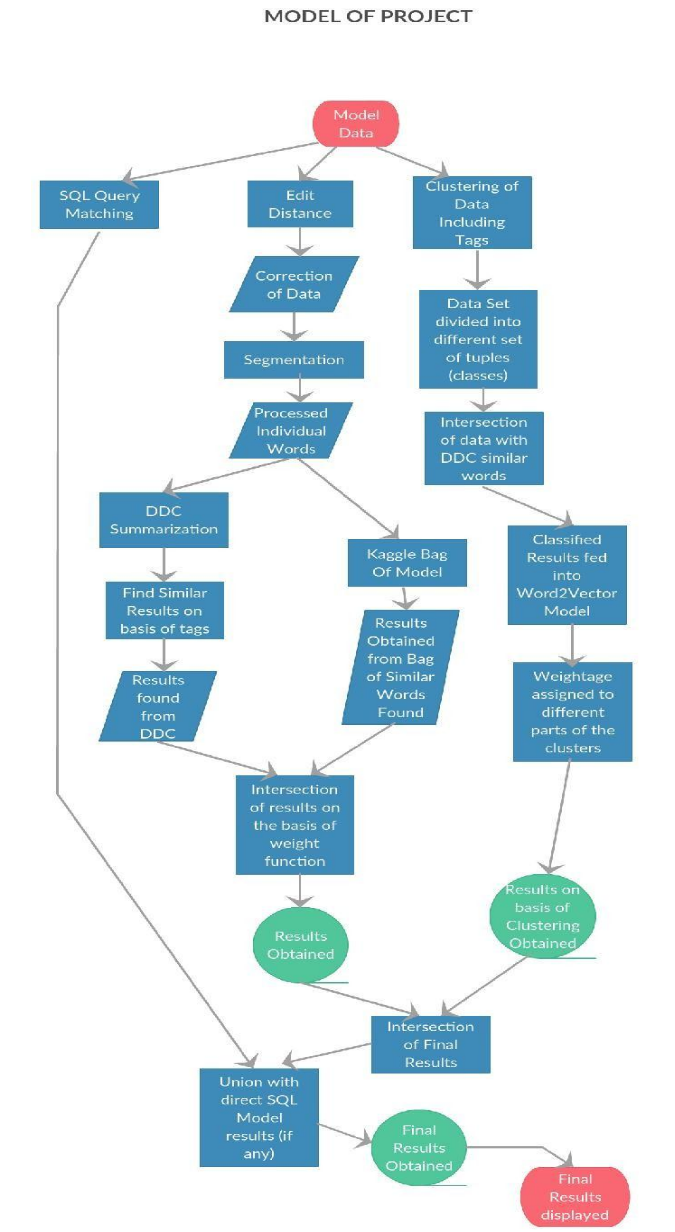

MODEL OF PROJECT
Model Data
SQL Query Matching
Edit Distance
Clustering of Data Including Tags
Correction of Data
Segmentation
Data Set divided into different set of tuples (classes)
Processed Individual Words
Intersection of data with DDC similar words
DDC Summarization
Kaggle Bag Of Model
Classified Results fed into Word2Vector Model
Find Similar Results on basis of tags
Results Obtained from Bag of Similar Words Found
Results found from DDC
Weightage assigned to different parts of the clusters
Intersection of results on the basis of weight function
Results Obtained
Results on basis of Clustering Obtained
Intersection of Final Results
Union with direct SQL Model results (if any)
Final Results Obtained
Final Results displayed

Figure 3. This shows the complete model of the project, how the data is preprocessed and how it is taken forward through different pipelines simultaneously and then aggregating the results based on some threshold and then they are ranked at last, thus, giving back the results.

In figure 4, the main part of the model in which most of the processing occurs is shown, it takes most of the time when executed on a user's query, this is the main pipeline through which the data flows and gives back the results.

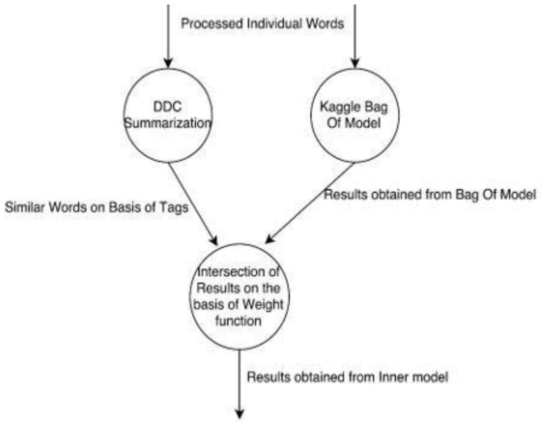

Processed Individual Words
DDC Summarization
Kaggle Bag Of Model
Similar Words on Basis of Tags
Results obtained from Bag Of Model
Intersection of Results on the basis of Weight function
Results obtained from Inner model

Figure 4. This shows how the data flows through the main model which is present in the composite model.

Flow Explained Thoroughly

Firstly, a user enters a query, then the query is corrected accordingly through the minimum edit distance module and is corrected if the words are not correctly segmented, they are segmented in the following module and thus, are corrected.

Now what comes out is the final user query, which is tokenized according to the word separators and each word is fed into 2 pipelines at the same time:

1. First Module consists of DDC Summarization, in which the classification of books has been done based on the DDC number and the results found similar to the query are given out from this module.
2. The second Module consists of the Kaggle Bag of Model; the Word2Vec model gives us similar words according to the neural network trained before.

Then, results found in the book titles are fed into a weight function, which uniquely identifies each book in the database and gives you the final results according to the weight function implemented.

There was another pipeline as well from the start which gives us the minor results but do gives us some, which is according to the clusters obtained on the data, these clusters do not change for each user's query and they are generated before and thus, just matching them with the clusters takes place in the pipeline and thus, these results are obtained and are passed through a second weight function, which is quite like the first one.

After all this, direct substring matching also takes place and those are amalgamated in the last phase of the implementation, which is again a weight function, which is mostly the same as before, just in this function, now ranking of the results is also done based on the score given to each result before and thus, final results are obtained coming out of this full model consisting of mainly 3 pipelines and 4 phases as explained before.

5. RESULTS

The results comply with the objectives of the project; we have tried to develop a generalized recommendation system and improved the search results at NNCL, Thapar University.

1. If a query is correct, it gives us the results having that keyword as well as the recommendation near to that query.
2. If the query is misspelled, then that query is corrected and then fed into the main pipeline giving accurate results.
3. If the query is not segmented correctly, then the query is segmented in the second module of the model and then is fed into the main pipeline, the results come out to be accurate as well.
4. If the query does not make any sense then no results are shown for that type of query thus, giving zero results.

The results are also ranked accordingly and thus, the books are shown according to the relevance of the user's query. As you can see in figure 5, that is the front end of the project and in Figures 6 and 7, it is showing the results for a query executed on the system having this model developed. The results can be compared as shown in figure 8, which shows the current results for that query, our model not only includes those results, but it also gives other possible recommendations and is accurately ranked as shown the figures. This justifies the composite model created for the search optimization and recommendation of the books at NNCL, Thapar University, and this model is quite implementable on any library around the globe as well, as it uses international standards, as well as the model, does not depend on the domain of the books anywhere. Ultimately, with the full model in place, we could achieve an accuracy of 85% which is a lot in the field.

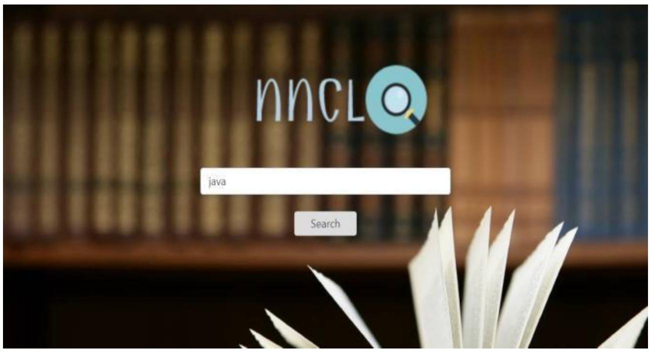


Figure 5. This is the front end created in Flask for the project to run on the servers.

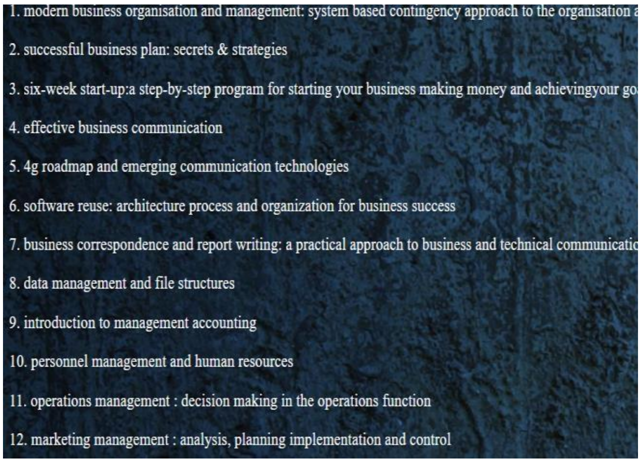
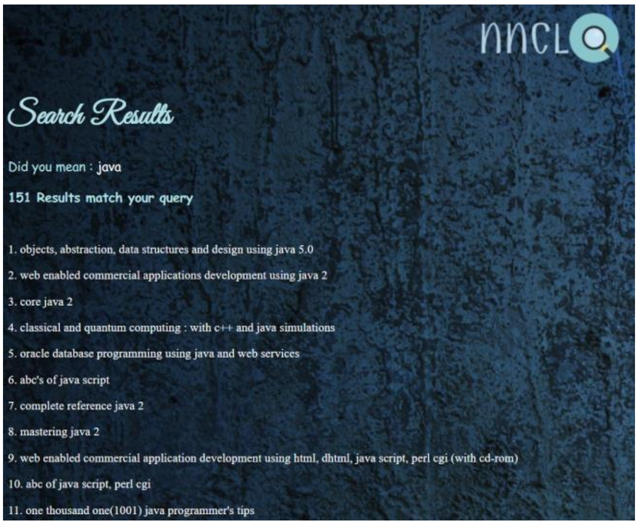

1. modern business organisation and management: system based contingency approach to the organisation a
2. successful business plan: secrets & strategies
3. six-week start-up:a step-by-step program for starting your business making money and achievingyour go
4. effective business communication
5. 4g roadmap and emerging communication technologies
6. software reuse: architecture process and organization for business success
7. business correspondence and report writing: a practical approach to business and technical communicatic
8. data management and file structures
9. introduction to management accounting
10. personnel management and human resources
11. operations management : decision making in the operations function
12. marketing management : analysis, planning implementation and control

Figure 6. Ranked Results for query "Business management"



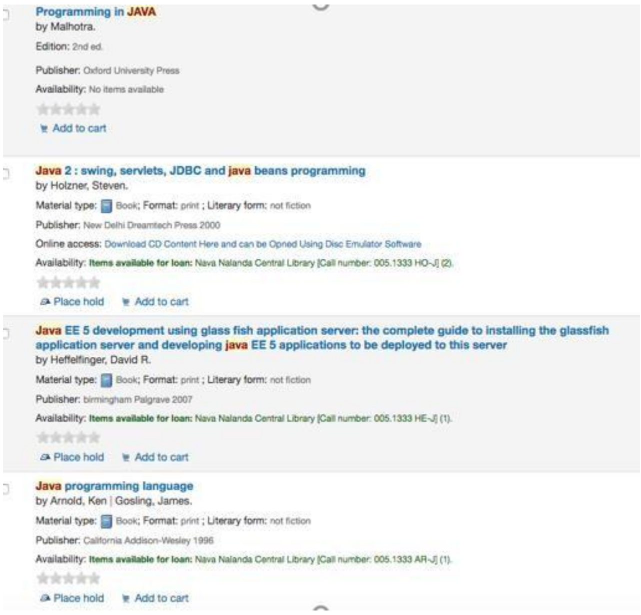
Search Results

Did you mean : java

151 Results match your query

1. objects, abstraction, data structures and design using java 5.0
2. web enabled commercial applications development using java 2
3. core java 2
4. classical and quantum computing : with c++ and java simulations
5. oracle database programming using java and web services
6. abc's of java script
7. complete reference java 2
8. mastering java 2
9. web enabled commercial application development using html, dhtml, java script, perl cgi (with cd-rom)
10. abc of java script, perl cgi
11. one thousand one(1001) java programmer's tips

Figure 7. Ranked results for the query run for "java".


Programming in JAVA
by Malhotra.
Edition: 2nd ed.
Publisher: Oxford University Press
Availability: No items available

Java 2 : swing, servlets, JDBC and java beans programming
by Holzner, Steven.
Material type: Book; Format: print ; Literary form: not fiction
Publisher: New Delhi Dreamtech Press 2000
Online access: Download CD Content Here and can be Opned Using Disc Emulator Software
Availability: Items available for loan: Nava Nalanda Central Library [Call number: 005.1333 HO-J] (2).

Java EE 5 development using glass fish application server: the complete guide to installing the glassfish application server and developing java EE 5 applications to be deployed to this server
by Heffelfinger, David R.
Material type: Book; Format: print ; Literary form: not fiction
Publisher: birmingham Palgrave 2007
Availability: Items available for loan: Nava Nalanda Central Library [Call number: 005.1333 HE-J] (1).

Java programming language
by Arnold, Ken | Gosling, James.
Material type: Book; Format: print ; Literary form: not fiction
Publisher: California Addison-Wesley 1996
Availability: Items available for loan: Nava Nalanda Central Library [Call number: 005.1333 AR-J] (1).

Figure 8. This figure shows the results of the current system at NNCL, which does only substring matching.

6. CONCLUSION

The whole project completed until phase 4 has been great learning especially in fields of Machine Learning, Natural Language Processing, and Software Engineering. The challenging problem to cater to requests needs by use of partial/incorrect data and to produce correct results led to iterative model development. Since the scope of the project was humongous it had to be done in a way so that the error rate is minimal at each stage. This was achieved by checking the performance metrics at every stage and performing the comparative analysis. Started with basics we developed a simple model based on Hidden Markov Model, N-gram technique However the results were not good enough when only the Spelling checking and segmentation were incorporated and more of background knowledge.

To overcome this shortcoming lot of literature and background were studied of the different library models and the present model of KOHA [10]. To make it more efficient the result filtering technique was altered and made us standardize the tagging generations using the DDC numbers and finding similar words using the Neural nets. The improved results were certainly motivating and close to the searched one.

We decided to further scale up our model which required a planned and exhaustive study into requirement analysis deep study such as the failure of the particular component, the progress that is done till now, the sequence in which work has to be done and the theoretical information about each component and aspect of the project. Later, we finalized how our product should like for the end-user i.e. design giving a lucid picture of the system we plan to create and the required functions that must be performed. The design specifications developed led us to identify the constraints of our product.

The most important part of the project was whether it could eradicate the current problems faced by the users on the current system. The key points which segregate our solution to the present solution are:

- Spelling correction if the user enters something wrong based on the dictionary generated by the library data which in the earlier system was not present.
- Segmentation of the concatenated words if the user has missed it by chance.
- Finding similar words using the Kaggle bag of models and giving recommendations to users of the various similar books that are present if any incomplete or partial data is provided.
- Aggregated results of DDC tags and built Neural Network gave more efficiency and filtered in a quick time which earlier took a lot of time.

7. FUTURE SCOPE

Since this project is quite large and challenging a lot of time has been given to acquire the knowledge background and the models which could be implemented to improve the current system.

Further work is also a rigorous development and integration task which is divided as:

- Integrating present SQL model to the newly build Machine Learning Model.
- If possible sending a patch to the “KOHA” community for an update.
- Using deep learning Recurrent Neural Network Model for better relation capturing.
- This model can also be extended for other languages in which the book titles can have words from other languages as well.

8. REFERENCES

[1] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. ICLR Workshop, 2013.

[2] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. Advances in neural information processing systems. p3111- 3119. 2013.

[3] Y. Bengio, R. Ducharme, P. Vincent. A neural probabilistic language model. Journal of Machine Learning Research, 3:1137-1155, 2003.

[4] Quoc V Le and Tomas Mikolov. Distributed Representations of Sentences and Documents. 2014.

[5] http://explorecourses.stanford.edu/ 5.

[6] http://www.puttypeg.net/papers/vector-chapter.pdf

[7] Wiener, Erik, Jan O. Pedersen, and Andreas S. Weigend. “A neural network approach to topic spotting.” Proceedings of SDAIR-95, 4th annual symposium on document analysis and information retrieval. Vol. 317. 1995.

[8] https://code.google.com/p/word2vec/

[9] https://www.oclc.org/en/dewey/features/summaries.html

[10] www.koha.org/