NCBI Document Recommender

Andrew Y

Figure 1. National Center for Biotechnology Information

Abstract

For students, finding relevant research papers for their studies is a problem faced in every assignment outside of their expertise. The goal of this project is to create a useful tool to find papers useful for their desired topic. Scope-wise, the model will work with documents from the National Center for Biotechnology Information, or the NCBI, as they have the an extensive database of documents relating to biology, mainly bio-informatics, bio-medicine, and bio-technology. The desired outcome is for the user to be able to input a keyword, and receive a list of documents that would be relevant to their work. To do this, our project will use the NCBI API to pull in documents from the database, and then utilize an unsupervised learning model to find a list of relevant documents to return.

ACM Reference Format:
Andrew Y. 2018. NCBI Document Recommender. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 3 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

The project is a recommender system model (tentative) where a user will be able to input a keyword to find a list of documents whose topics are relevant to the keyword. The problem this model attempts to solve is to help minimize the effort required to find useful sources or references to use when one is attempting to do educational work outside of their field of expertise. For example, if a student was writing an essay about different allergies for a class, then they would be able to enter the keyword "hay fever" and be able to receive numerous published educational pieces around hay fever. While not life changing, this would help reduce the time spent browsing numerous documents and allow the user to allocate more time to other tasks of their assignment. To help keep the scope manageable, the project will be working with

Conference'17, July 2017, Washington, DC, USA
© 2018 Association for Computing Machinery.
ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

data from the National Center of Biotechnology Information, or NCBI for short. They are a research institute that is a branch of the National Institutes of Health, and is known for housing a multitude of databases containing tools and services relating to biotechnology and biomedicine. They also house databases dedicated to bio-related literature, which is what this project will be pulling the data from.

2 Related Works

As a recommender system, it can utilize obvious references to the renown Google search engine. In addition, NCBI already have their own search function for documents, a tool named Entrez, which is their dedicated text retrieval system. However, their existing search function only has four different ways to sort their result, which are article order, publication date, journal number, or PMC live data. The goal of the project is to create a different way to receive the results, one that utilizes the document contents rather than their metadata which is different from the existing methods. In doing so, we can ensure that the resulting list of documents will be of more use to the user than scrolling through documents listed in chronological order.

3 Proposed Work

The way the project will be tackled will be to first utilize our main tool Entrez to pull a data set from the database. In regards to data warehousing, fortunately enough the NCBI organization does a perfect job of keeping the data well managed in their own "warehouse" so the data can be kept in its original format. This data set consists of individual documents as its rows, and document features as its columns. The document features mostly consist of metadata about the document, such as publication date, the journal that it belongs to, or the author. From these features, there are three main ones that will have the most focus: document abstract, document reference number, and document author. Document abstract is our biggest priority as it will be where the bulk of the modeling will utilize. The abstract is used over the actual text data as it is more feasible in regards to performance, and also helps combat over-fitting. This is due to it acting as a summary for the actual article text, and thus can be treated as a form of dimension reduction. As the abstract data is still also in text form, we will undergo the typical text pre-processing steps such as but not limited to, removing accents, word lemmatizing, and removing stop words. Once the pre-processing has been completed, it will be fed through a TF-IDF Vectorizer model to create a document-term matrix for use as an input for our model.

The document reference number feature is a count of how many times the document has been referenced by other articles in field. It will be utilized to help act as a metric for determining a document's significance or quality, as documents that are highly referenced are most likely well-written or well-researched enough if they are frequently referenced by other members in the community. However, there is an issue of a significant amount of documents not having a reference number; as not having a reference number is not the same as having a reference number of zero, we do not want to leave out all of these documents. Thus a regression model will also be created to predict the supposed reference numbers based on their abstract contents. By using the reference number for each document, it can be used to prune the documents that might not be as helpful if a threshold is set to only keep documents with a reference number above a certain amount.

The last kept feature is document author. This feature isn't particularly used at the moment but potentially could be used to help find similar documents if an author can be considered well-known in the field. This is a metric that will be revisited later if time permits.

Once these tasks have been completed, then the actual modeling can start. The current plan is to utilize a clustering model, in which similar documents will be grouped together based on their abstract's data in the document-term matrix from the TF-IDF Vectorizer. The size of the cluster can be preemptively matched to be the same as the list of documents that we desire to return. To choose the cluster of documents to return, the metric for decision will be delegated to the reference number yet again. If the sum of all reference numbers in a particular cluster is the highest out of its peers, then it makes a solid argument for that cluster containing the objectively most useful documents.

While the regression and clustering models have not been completed yet, once they are finished it will then be time for hyper-parameter tuning to help improve our model's performance before it is considered finished.

4 Evaluation

For evaluations, in terms of the regression model, the metric that will be focused on will be Poisson Loss Regression, which is a metric that can be found from the 'statsmodels' Python library. The main advantage of this metric is that it is commonly used for count data, such as how much an item is predicted to be sold during a seasonal period, and this matches exactly as our target feature, the reference number which can be considered a count of how many times the document has been referenced by other sources. For the model itself, will attempt the usual regression models such as linear regression, poisson regression, decision tree regression, and random forest regression.

In terms of the clustering models, the metric used will be the silhouette score, which is the metric for determining how compact the cluster is and the calinski harabasz score to view model performance. For the model itself, will experiment with both hierarchical clustering and K-Means clustering though most likely K-Means will out perform due to our ability to preemptively set our desired size for the list of documents, which can be used as the 'K' for the K-Means model.

5 Discussion

Proposal Stage:

The first priority of the project timeline is to get the data ready. This will involve utilizing the Entrez API to pull data from the database, and doing extensive cleaning of the text data to better prepare it for modeling. The second milestone will be to ensure that our metric of determining whether or not a document can be considered significant enough to be considered for returning. The third milestone will be to be able to create a model that will return a list of significant documents. Once theses steps are finished, we can undergo model optimization and performance tuning.

The potential challenges mainly resolve around evaluating the recommender system, in which the clustering method will act as a suitable backup plan as a way of improving performance.

Project Checkpoint Stage:

The project is proceeding smoothly, the time spent working with the data after the project proposal gave ample time to test different theories and potential problems. The data for the model that utilized the database site's tool that pulled the document data using the API has been acquired both consistently and reliably. The data wrangling has been completed as well, with the text data of the abstract cleaned into a way that best suits the TF-IDF Vectorizer input. The TF-IDF Vectorizer as well has been used to create a document-term matrix ready and prepped to be used for modeling.

Almost all of the problems that popped up during the proposal brainstorming stage have been dealt with. For the issue of finding a way to evaluate a recommender system when there is no feedback to improve off of, it was swapped to utilizing a clustering method instead that judged the documents based on their sum of reference numbers.

However, an unforeseen issue did appear when Exploratory Data Analysis was done as it turns out a significant number of documents did not have a reference number for the model to work with. As a missing reference number was different than documents that had a reference number of zero, simply leaving out these documents would be a waste. Thus came the idea of creating a second model, a regression one, to help predict the count of the reference number if the number was not missing.

As the modeling portion is still a work in progress, it will be discussed in the following report.

6 Conclusion

In the best case scenario, the project will end up as a useful tool for users that are undergoing biology related studies, where they can simply enter a keyword and receive an extensive list (or short list depending on the user's wishes) or potential documents for the user to study and utilize for their work. < Add key findings later > < Add future work later >

7 References

Entrez's Bio Python Package
https://biopython.org/docs/1.76/api/Bio.Entrez.html