Curating Community Spectral Libraries Using a Recommender System

Nikolaos Koulouris
A53087405
nkoulour@cs.ucsd.edu

Benjamin Pullman
A53100329
bpullman@cs.ucsd.edu

1. BACKGROUND

Global Natural Product Social Molecular Networking, or GNPS, is a resource for sharing and discovering mass spectrometry (MS) data for natural products [2]. Natural products, which are simply any substance that is found in nature, are pervasive in life and hold tremendous power in their study and applications. As an example, the 2015 Nobel Prize in Medicine was presented to a pharmacologist, Youyou Tu, who used a natural product derived from wormwood that proved as an effective antimalarial [1].

For the purpose of this paper it is important to point out that GNPS has two distinct types of libraries: those that are from third party sources and the GNPS library that is community curated. In later analysis we will define how these differ, but we will define GNPS to only contain spectra in the GNPS library unless otherwise specified.

Using GNPS, scientists around the world can upload data from their experiments and both contribute and help curate the repository of natural products MS data. However, as the data is uploaded from various sources, it requires some sort of curating to ensure that all the data is accurate and legitimate. One way to show this is through standard peer review and publishing the results. This process often requires a long turn around, and given the sheer amount of spectra, this is not always possible.

The other method of curation is by user annotation. This gives the data legitimacy by allowing peers to review and comment on spectra a few at a time, then when other users see the annotations, they are confident they are correct. The goal of this paper is to help facilitate user curation by providing a recommender system for users to show them spectra to consider annotating.

2. RELATED WORK

Our approach is to create a recommender system that inspires people to contribute to a scientific community. We want to take advantage of the social aspect of our system, while making it as relevant as possible for our users and attractive to new users.

One similar approach appears in a recommender system for academic papers [3]. This system uses social features to hone a personalized approach for users, usually academics, to find papers to read that they are interested in. One key finding was a metric which defined similarities between papers, which often would recommend papers from researchers to their PhD advisors. We hope to do similar in our model, recommending spectra that are extremely close to the users research, as these we believe will be more interesting and easier for the user to annotate.

Another key insight for our paper comes from a study about Wikipedia community maintenance [4]. In their paper, they show that there exists a significant gap between the maintenance of some articles and the popularity of said articles. They go on to show that almost half of Wikipedia articles viewed are not aligned in quality with the number of views each article gets. This was helpful as we needed a metric to help define what constituted something worthwhile to review.

3. APPROACH

To create a recommender system for GNPS annotations, we first consider the data we were able to scrape from GNPS. From there we can define a predictive task so we can concretely define success in creating the recommender and then define a model.

3.1 Exploratory Analysis

Below are the four datasets we created from the data on the GNPS servers. Users and datasets were the simplest to scrape, each just being a call to the server requesting a json file with the respective data. To get the results information, we had to crawl a folder containing all the data from the jobs run by each user. The folder had other folders representing the actual outputs themselves, and to combine this information we created a hash table that linked the usernames to the results information. To get the spectrum information, we took a union of each result (irrespective of user) to create a list of unique spectra. From that list of unique spectra, we wrote a script that sent a request to the server for each spectrum individually. From the individual spectra we filtered out server side errors and binned them into a series of json files. Since all our data was less than 10GB we had no problem doing everything in memory.

3.1.1 Users

There are 375 registered users in GNPS. Users data unfortunately does little more than indicate how many there are. Of these 375 users, 275 have searched data in the GNPS library, and these are who we'll focus on.

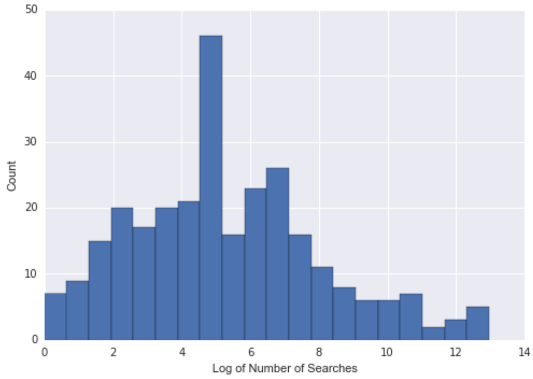

Figure 1: Searches per user in GNPS

3.1.2 Datasets

There are 6,787 submitted datasets. Information about the datasets include the workflow that was run, the time it was run, if the data contained is public or private, and other features which are specific to the dataset but will not be used in our analysis. Of the fields contained in each dataset, there is a mix of categorical data correspoding to the species and instrument as well as freeform text in the titles and descriptions. Each dataset additionally had a link to the results after running the dataset through a defined workflow.

3.1.3 Results

These are the identifications from users searching mass spectrometry data, either datasets they submit or running queries on datasets that are already uploaded to GNPS. There are 12,713 distinct tasks we are examining leading to 5,719,677 different results. Each result contains a spectrum as well as metadata about the run and result, including its Pubmed identification, if it has one, the PI of the lab that ran the experiment, and the quality of the library the spectrum was from.

3.1.4 Spectra

This is the data about each unique spectrum of which there are 400,958. For each spectrum we see metadata including the current annotations (user identifications and notes about the spectrum), the library membership of the spectrum and some physical data about the spectrum itself, and the number of distinct peaks.

From this data we hoped to filter everything. Looking specifically at the GNPS results, we notice that each user has a fair number of searches.

3.2 Network Representation

As a way to better understand the relationships between users, we constructed a graph linking each user where the edges are defined by a set number of common searched spectra. This allows us to understand how similar the searches are between users and ultimately help us figure out what the users might be interested in annotating. We wished to pick a network representation that would mimic the users actual interest as well as potential publications.

In defining this network, we needed to consider what constituted a similarity between each user. We defined this threshold using two ideas:

Relevance. We want to take the top-k spectra searched for a user. This way we can represent exactly which spectra the user is likely to be most interested in. This approach has the harm of potentially including contaminants (spectra which are not interesting but rather an artifact of collection), but for our purposes, we assume these to be filtered by GNPS.

Connection. How many relevant spectra two users had to have to constitute a similarity between two users.

By combining the two metrics, we define how many of the most relevant spectra have to be in common in order to define a relationship between two users with the goal being to construct a network that was as connected as possible so we could always make a prediction from it, but also sufficiently sparse so that every user did not appear to be interested in the spectra of every other user. We also wanted to make our network robust such that if we added another user, that user would somehow be connected to another user.

We ended up using a connection metric where we examined the top ten searched spectra per user and set it so only one spectrum had to be in common between two users' top ten spectra. Please see Figure 2 for a representation of the graph. As is shown in the diagram, the graph is highly connected, with the average number of neighbors being 29, which out of all 275 users ends up being a little denser than we expected but this does not effect the result. What is also important is the maximum number of neighbors a node has is only 76 implying that no user dominates and every user can relate to any spectrum through that user.

From this network we would create a metric to link users and spectra. This would be a breadth first traversal of the user graph, searching each users relevant spectra for our target spectrum. The depth of the traversal, and the number of relevant users at the shallowest point at which the target spectrum is found would define this metric.

3.3 Predictive Task

Our goal was to find a way to curate the data as part of the GNPS library. To do this, we built a recommender system that finds the most appropriate spectra to suggest to users to annotate. We wanted to find a quantifiable way to state the predictive task based on the data available through the GNPS system and a task that would have a measurable result, so that we could evaluate our approach. Below is the predictive task we identified.

Since our main goal is to curate the data that are part of the GNPS library, we made assumption that spectra that are part of other well known libraries do not need to be curated. So, we split the spectra that belong to the GNPS library in two sets, the training and the test sets.

There is a need to curate all of the spectra in the GNPS libraries, but because of quantity and work that is required from the users, only a few at a time can be curated, especially as a first metric. Our goal was to add a legitimacy to the library so we make the assumption that the ones that are

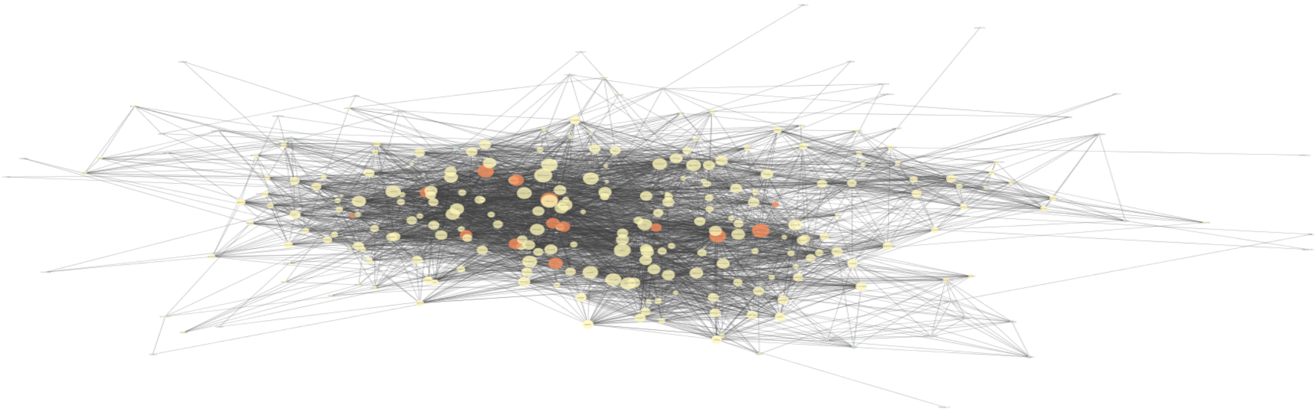

Figure 2: Users in GNPS

most searched for by users should be curated first. Our predictive task is simplified into predicting, given a user and a spectrum, how many searches for that spectra will take place.

The real challenge of the task was to identify features from the data that are available that are capable of predicting what spectra people are searching for. We employ mostly metadata from the results and spectrum information but additionally make use of one physical metric, the number of peaks. Due to the limitation of explicit features available in the metadata for the spectra, we decided we needed a network representation to account for the social nature of GNPS.

In addition, although we could evaluate our performance based on the MAE or the RMSE we decided that we needed a different metric that would be more suitable in our specific problem. That is because when this recommender system is going to be integrated into GNPS, only very few spectra are going to be recommended for curation to each user due to UI – our goal is to cleanly present a few spectra for the user to annotate. As such all we need, is at least one of those few specta to be accurate so that the user chooses to annotate it. As a result, we chose a variation of the accuracy (at 3, 5, and potentially 10) as a metric of the results.

3.4 Models

To build our recommender system we first considered all the different models we could choose.

Our first option was a simple collaborative filtering model. However, this model does not seem good for our specific dataset because we would face the cold start problem when making our predictions on the unseen data. Since the model doesn't allow for any features to be added, it would not work well in the unseen part of the data that we want to make predictions on. And it would be more difficult to incorporate the social metric of a graph that is very intuitive in making such predictions.

Our second option was a linear regression model. This is the model we chose to implement. With this model we could use features that appear in the dataset that we used and also incorporate other metrics, such as a user bias and a network bias. Our linear regression model follows.

$$f(user, spectrum) = \alpha + \theta \times x_{spectra} + \beta_{user} + \psi_{user,spectrum}$$

where α is the offset, $x_{spectra}$ is the feature vector, β_{user} is the user offset and ψ is a metric defined from the user graph above and is of the form.

$$\psi_{user,spectrum} = \frac{1}{|N_{user}|} \sum_{u \in N_{user}} \begin{cases} 1, & \text{if } spectrum \in u_{spectra} \\ 0, & \text{otherwise} \end{cases}$$

where N_{user} is the set of all the neighbors of the user as defined from the graph and $u_{spectra}$ are the spectra of each neighbor user u.

3.4.1 Features Selection

First we considered all the different features that could be found from the available data. Some of them worked and improved the performance of the model and some of them failed to identify the most searched spectra.

- The number of annotations a spectrum has received is a feature that helped improve the model. In essence, spectra that have been annotated many times, are more likely to be searched for, and by our assumption, annotated in the future.
- The existence of a PubMed ID for a specific spectra helped identify the most searched spectra. Intuitively, a PubMed ID helps to verify the spectra.
- The number of spectral peaks. The more peaks a spectra has, the more complicated it is. So, given that the theta value of the model for this feature is positive, it means that complicated spectra tend to be more searched for.

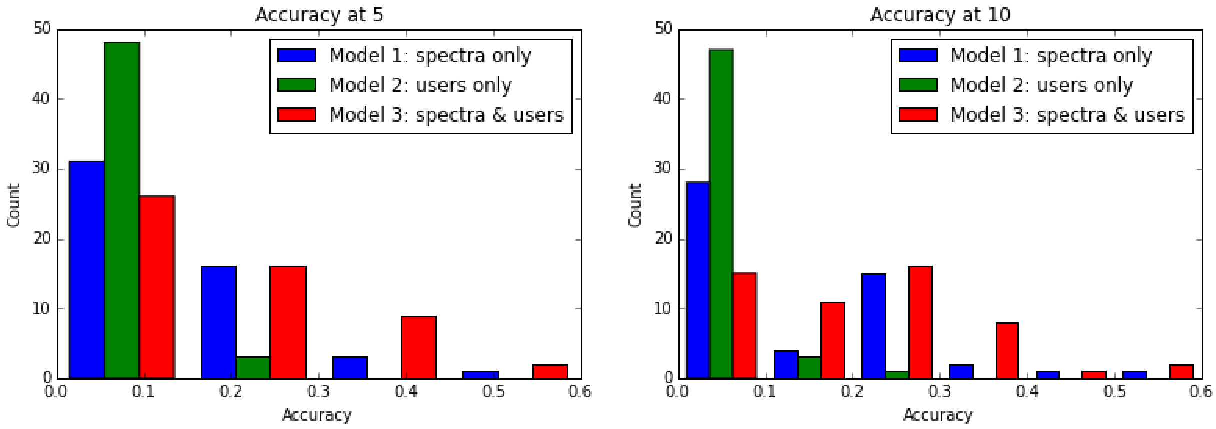

Figure 3: Accuracy of all the models

- Each spectrum's status, reported quality, and charge are some simple features that didn't improve the performance of our model.
- We also tried adding all the different categories of the data collectors and the library classes as categories, but this features didn't improve the performance either.

3.4.2 User Bias

All the simple features described above were not enough to make a good model. Next, we added a user bias. There are some users that are more active on GNPS and search for more spectra and the model takes that into account by adding as a feature the average searches each user has done on the training set.

3.4.3 Network Bias

We added a feature based on the network that was previously described. For each pair of (spectrum, user) that exists in the dataset, we get from the graph the number of neighbours of the user that are interested in the specific spectrum.

4. RESULTS AND DISCUSSION

To measure our results and our model, we used the accuracy at 5 and at 10. The accuracy is a suitable metric for our model because we want to recommend at least one or two spectra to each user that are relevant. In practise, even if every user annotates one of the spectra that we recommend, that will be a big success over the current state of the system. We chose accuracy at 5 and at 10 because at the UI of the system there is only limited space to make suggestions.

We assume that to each user the top ten percent of his most searched for spectra are interesting. So, we measure how many of the the spectra that each user searched are in the ones that he is interested in. We also only considered users with more than 5 distinct spectra searches in the test set. We tested on one tenth of the GNPS dataset that we withheld from the training process of the model.

In order to validate that a user,item model was necessary, we considered three differing models. Model 1 is the model that consists only of the features that we derived from the information we had for each spectra. Model 2 is the model that considered only the users' information, i.e. the user bias and the network metric. Finally, model 3 is the combination of the two previous models.

As it can be seen on Figure 3, our model seem to significantly outperform the two other baseline models at both cases.

5. FUTURE WORK

The recommender system described in this paper only scratches the surface for what we hope to do in GNPS. Our next steps are to incorporate our model into the live GNPS website and to begin to collect impressions about how it works in practice. From there, we can collect data on if the spectra are interesting to the user and more directly asses the system. By actively recommending a spectrum for a user to annotate, we can then have both positive and negative feedback to train on.

6. ACKNOWLEDGEMENTS

Ming Wang helped us with some of the initial access to data and help scraping the datasets as well as provided some initial direction for our approach.

7. REFERENCES

[1] E. Callaway and D. Cyranoski. Anti-parasite drugs sweep nobel prize in medicine 2015. *Nature*, October.

[2] http://gnps.ucsd.edu/.

[3] J. G. K. Joonseok Lee, Kisung Lee and S. Kim. Personalized academic paper recommendation system. *SRS'15*, 2015.

[4] L. T. Morten Warncke-Wang, Vivek Ranjan and B. Hecht. Misalignment between supply and demand of quality content in peer production communities.