Promoting Growth of New Stack Exchange Sites

David Martuscello, *NYU*
Thang Chu, *NYU*

***Abstract*—Stack Exchange is a collection of question answering websites with a unified structure each focused on a specific topic. Users in the Stack Exchange community can create new websites as long as they can draw enough people in to support the sites. The goal for this project is to provide an application for the clients to grow their new website based on targeted marketing. A client will use the application to sort through users of Reddit and other Stack Exchange sites and send them a message invitation to sign up on the client's new website. This targeted marketing approach will allow the client to reach more potential users, and increase the speed and effectiveness of their outreach campaign.**

***Index Terms*—Analytics, Spark, Big Data, Machine Learning, Stack Exchange, Reddit.**

I. Introduction

STACK Exchange is a collection of question answering websites with a unified structure and a single user base. Each site has a focus that the questions are based on. Since each site relies heavily on the vast amount of user's knowledge and demand, it is essential for the website owner (the client) to quantify such data. The goal for this project is to provide an application for the clients to grow their new website based on target marketing and relevant cross-site promotion. A client will use the application to identify the potential existing users on other Stack Exchange sites and Reddit and send them a message invitation to sign up on the client's new website. Moreover, our application will also identify Reddit posts that contain similar questions on the client's web site and send this information to the client. This way, client can directly promote her website by linking the relevant Stack Exchange post to the Reddit post. This will help to attract new users to visit the clients' website.

The project is carried out in two phases. In phase one, we use natural language processing techniques to build a content vector for each user including all the keywords for each topic. The data are Stack Exchange and Reddit comments across all existing sites and subreddits. In phase two, we build an application to identify potential users both from Stack Exchange and Reddit based on the topic that a client provides using Cosine similarity measure. Moreover, the same approach can be applied to identify existing Stack Exchange posts similar to the Reddit questions. We use Scala Spark to process data and build the machine learning model.

II. Motivation

The main motivation of the project is to find a more effective way for a website owner to promote their proposal on Stack Exchange. Since this website is only in the proposal phase, it is often difficult to estimate the user's demand. New users typically do not create account and start contributing until the website is already functioning. Due to the relatively large domain and related topics, it is also difficult to identify the potential users of the new website who are already existed inside and outside of Stack Exchange. Because of the similarity between Stack Exchange and Reddit, an existing Reddit user is likely to become a Stack Exchange user provided that the topics and questions are of her interest. As such, our application can create a significant impact for the clients in promoting their new website.

III. Related Work

Previous research mainly focus on different techniques to detect online community through social networks. In the research paper by Peter Baumgartner and Nicholas Peiper [1], the authors utilized big data and Twitter to discover emergent online communities of cannabis users. The authors target Twitter accounts that follow the validated accounts of 6 dispensaries in Oakland. Data collection is divided into 2 stages: stage 1 is collecting all accounts that directly follow the 6 dispensaries and stage 2 is collecting all accounts that follow the accounts found in stage 1. Data pre-processing is performed by removing accounts that only follow one other account in the network sample and the protected accounts. These accounts are considered noisy data in terms of community detection. For modeling, a nonparametric probabilistic model that optimize for statistical likelihood is used. Specifically, a hierarchical stochastic block model is used. To describe the communities derived from the model, a multimode content analysis is conducted. Each block is examined and labeled with 5 codes (or keywords) from a collection of 60 core codes. The blocks with spam accounts are also classified using metrics similar to Twitter Audits metrics. The same coding scheme is applied up the hierarchy levels. To identify types of cannabis consumers, blocks at level 3 are profiled. Only blocks that contain consumers in the illicit, recreational, and medical blocks are selected.

This study is relevant to our project because it presents a way to profile community based on the content associated with each user. In our project, each user is associated with questions/posts which contains tags/keywords. Using hierarchical stochastic block model, we can create blocks of user based on common keywords/topics. We can then go up the hierarchy level and evaluate the points at which the community or domain website is officially created. The analysis done on the Stack Exchange data can then be done on Reddit comments data to estimate the external interest.

The paper by Fortunato and Hric [2] provided a high level view of the current state of community detection. This paper focused on the concept of community detection. Communities can be detected within a network or graph. Currently, the exact definition of a community is not well-defined, but there are concepts that are common to community detection. The three main ways of considering a subgraph is by its internal connectedness (cohesiveness), or by its external connectedness (separation of the subgraph) and by combinations of the two. An important component of community detection is the validation step where the (typically unsupervised) clustering component is compared against benchmarks to see how well it finds the actual communities. This can be done on real datasets where the community structure is known or artificial benchmarks such a metrics for evaluating the communities that have been found. To assess the significance of the clusters found, the graphs can be compared to a null model. The paper then compares an array of community detection algorithms at length and concludes that the best algorithm depends specifically on the domain and problem.

A study by Preeti et. al. [3] proposed a community trolling approach based on active learning to help improving topic-based community detection. Community trolling helped filtering big, polluted data into smaller, unpolluted data. This was done through active learning which samples only a small set of data that provides the most relevant topics instead of applying topic modelling on the whole dataset. The study was conducted on hate data captured from Tumblr. After tokenization, data was vectorized using tf-idf and word2vec for comparison. Uncertainty-based sampling and density weighted sampling was in active learning to learn the most and least informative attributes. Training data was a small subset of labelled data using domain knowledge and Support Vector

Machine (SVM) was used as underlying classification model. After going through community trolling, only the most relevant data remained which was then used to classify topic-based community using Latent Dirichlet Allocation. The result showed that better topics were generated from the filtered data using community trolling. The SVM classification using the filtered data also outperformed the baseline SVM. As a result, community trolling using active learning helped removing noise from big volume of data, resulting in better analysis.

This study is applicable to our project because it proposes an approach to reduce the volume of data through data validation. It also touched briefly on LDA which is a technique to generate topics for topic-based community detection task. Moreover, the study is done using Sparks which is the same platform that being used in our project.

Latent Dirichlet Allocation (LDA) is one way to do topic modeling as described in [4]. Topic modeling is useful for taking documents and separating the words from this document into topics as a way to classify the documents into categories. LDA is a probabilistic model, meaning that the data is assumed to be generated from some probabilistic process where there is some hidden structure. The job of the model it to infer this hidden structure from the data. When new data is presented to the model, the model will indicate the topic that the data point fits best within. Each topic will have different words, each of which has a different probability of appearing in the document. Each topic is essentially a distribution over a fixed vocabulary. Each document will be a combination of different topics. The generative process for each document is to 1) choose a distribution over topics 2) For each word draw a topic from this distribution 3) Look up the terms associated with that topic and then draw a word from those terms.

IV. Application design

Our application is designed in two parallel tiers that reveal slightly different insights about our data. Both of these sets of insights work towards the unified goal of promoting growth of new Stack Exchange websites.

Both approaches use a ”PROPOSAL” file as a representation of a new website. This data was taken from the staging site for new Stack Exchange websites known as ”Area51”. The example questions from this proposal are intended to define the topics of the website so this is a realistic experiment because it simulates the information that would be available when a website is early on it its growth phase. The text of the PROPOSAL file will be compared to the individual users of each website in order to obtain a similarity score for each user. The hypothesis of our project was that the similarity in vocabulary between the users and the PROPOSAL would be a rough proxy for their knowledge in the topic of the website. This similarity score is the ranking that will be used to determine which users will be marketed to by the website owners via direct messaging or email.

The first approach looks at Stack Exchange alone and tries to find users within other parts of the community that would also be good candidates for the ”target” website. This is an important source of users because they are already on the website, therefore their barrier of entry is lower. However, given the strict divisions of topics on the Stack Exchange websites, it is likely that most users are already in the communities that best match their interest. This is why another dataset was needed to try to bring in users from another source.

The second approach analyzes the users of the website Reddit. Reddit has a diverse array of groups and users making it an interesting place to look. Additionally, it often functions similarly to Stack Exchange as a location for questions and answers from experts, making it a natural source for people with inclinations for question answering. The

Our project uses TFIDF and LDA techniques to generate lists of these users who will be a good match for the PROPOSAL website. These techniques and their application are discussed in the Experiments section.

Figure 1 shows the project diagram. The lower portion of the diagram illustrates a separate task that is discussed in the Future Work section.

Figure 2 shows a prototype of the user interface. A simplified version of the ”individual outreach” section was actually implemented for this project. The right portion of the diagram illustrates a separate task that is discussed in the Future Work section.

V. Datasets

Reddit comment data 1.1 TB containing all posts and comments on the site between December 2005 and November 2015. This data is collected from Pushshift.io and stored on NYU High Performance Computing clusters. Pushshift.io is maintained by Jason Baumgartner and contains various articles relating to big data, social media ingest and analysis and general technology trends.

The Stack Exchange dataset is made up of various xml files that contain all the data from the Stack Exchange collection of question and answer websites from 2015 to present. The files provided for each Stack Exchange site are Users, Posts, Tags, Comments, and Badges. The data was processed in order to select the text that would be relevant to a particular user in order to associate them with the text of a particular proposal. Several different text features (Posts, Comments, and Profile Descriptions) for each user were combined into a single string and then converted into vectors that represent the vocabulary of a particular user. This allowed them to be compared to the PROPOSAL website based on vocabulary.

VI. Remediation

Our application matches Reddit users to the new site proposals by comparing the vocabulary of the user with the proposals to determine if a user could be a contributor to the new website. The client can then decide to directly message the users found by the application. To facilitate this intervention by the client, our program provides a ranked list of users along with links to the direct message page for that user (Figure 6). This allows the client to only send messages to the users that are most likely to be interested in this website. This would be very useful for a client looking to grow their userbase with minimal amount of wasted effort, by ensuring that every message is sent to a likely candidate.

VII. Experiments

Invalid entries such as deleted users, bot accounts, and empty posts were removed. The posts were then aggregated based on usernames which are unique in each dataset to create the the data matrices. The proposal text is also concatenated to the respective matrix (with username "PROPOSAL" for the Stack Exchange and username "[deleted]" for the Reddit data matrix). For each dataset, we generated a Term frequency-inverse document frequency (TF-IDF) word vector for every user representing all of his/her posts using Spark MLlib's RegexTokenizer, StopWordRemover, CountVectorizer and Inverse Document Frequency (IDF) functions. After that, the project was carried out over two iterations.

In the first iteration, we used the TF-IDF vectors to directly compare each user's vector and the proposal vector using Cosine similarity. However, the similarity score between the user's vector and the proposal vector was very low (close to 0). This is because each user's vector was represented by 180,000 words which made the overall data matrices very sparse. The vocabulary of the dataset was generated based on the words that appeared in at least 5 different users' post. Moreover, only users with significant content (defined by the length of theirs vector after the tokenization step) could contribute to the overall vocabulary.

As a result, we tried to mitigated the sparsity problem by re-iterating our experiment using Latent Dirichet Allocation (LDA) model instead of raw TF-IDF vectors. LDA is an unsupervised clustering algorithm which infers topics from a collection of documents. In our study, each user was treated as a separate "document". We trained our LDA model to find 160 topics over 10 iterations. This allowed us to represent each user's vector by the topics and topic distribution which were much denser. As a result, the similarity scores between the proposal vector and user's vectors were much higher.

To test the "goodness" of our analytic and validate our model, we used two different established websites and compared them to our proposal website. The PROPOSAL dataset was built with the example questions taken from the "Area51" website staging area for the site topic of Medical Sciences [LINK]. This was intended to simulate a new website in the "definition" phase of website development when only example questions are available. The two websites that were chosen as comparisons were the "Woodworking" Stack Exchange and the "Health" Stack Exchange. These sites were chosen because, when considering the Medical Sciences topic, Health is much more similar than Woodworking and thus should produce users with higher similarity scores. Indeed these results were achieved, confirming that our system does provide a reasonable ranking of users. These results can be seen in Figures 3 and 4.

VIII. Conclusion

The goal for this project was to provide an application to aid a Stack Exchange user in growing their new website. The application allows targeted marketing towards users of Reddit and other Stack Exchange sites who would be good candidates for the new site. Our implementation used several text processing approaches such as TFIDF and LDA to compare the vocabulary of users to the vocabulary of the PROPOSAL site. This comparison generated a similarity score for each user, leading to a ranked list of users that should be targeted. As a remediation step, the client is provided with a link to contact these users and attempt to bring them on as user of the new site.

Images of the final state of the project can be viewed in figures 3 through 6 as well as in the html files on github [LINK]. The project was successful overall, achieving the desired objective without many of the additional features that were initially imagined. There were several problems that we ran into including difficulty with text lemmatization, problems optimizing large RDD operations, and generating useful visualizations. Given these time-consuming obstacles we were pleased that we were able to achieve our basic objective on time.

IX. Future work

There are many different directions that could be taken with future work on this project. This section will discuss a separate idea that was considered given this same data. Some elements of this approach are shown in Figure 1 and 2 above.

The idea is to search through Reddit discussions and look for questions. The model will then search for this question on Stack Exchange to find an appropriate answer. The best matches will then be surfaced through the interface. It will then give the user the option to automatically post a link to this answer to the Reddit page. These links will act as a inroad to the Stack Exchange website, allowing people on Reddit to be exposed to this particular Stack Exchange community, helping it grow over time. This concept is included in the Design Diagram (Figure 1) and the User Interface (Figure 2).

Acknowledgment

The authors would like to thank Jason Baumgartner for making the Reddit data available, NYU HPC for providing the the data storage and analytic platform, and to Professor McIntosh for her guidance.

References

[1] P. Baumgartner and N. Peiper. *Utilizing Big Data and Twitter to Discover Emergent Online Communities of Cannabis Users.*, Substance Abuse: Research and Treatment, vol. 11, 6 June 2017, p. 117822181771142., doi:10.1177/1178221817711425.

[2] S. Fortunato and D. Hric. *Community Detection in Networks: A User Guide* https://arxiv.org/pdf/1608.00163.pdf

[3] G. Preeti, et al. *Community Trolling: An Active Learning Approach for Topic Based Community Detection in Big Data.* SpringerLink, Springer Netherlands, 10 Aug. 2018, link.springer.com/article/10.1007/s10723-018-9457-z.

[4] David M. Blei, Andrew Y. Ng, Michael I. Jordan *Latent Dirichlet Allocation* http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf

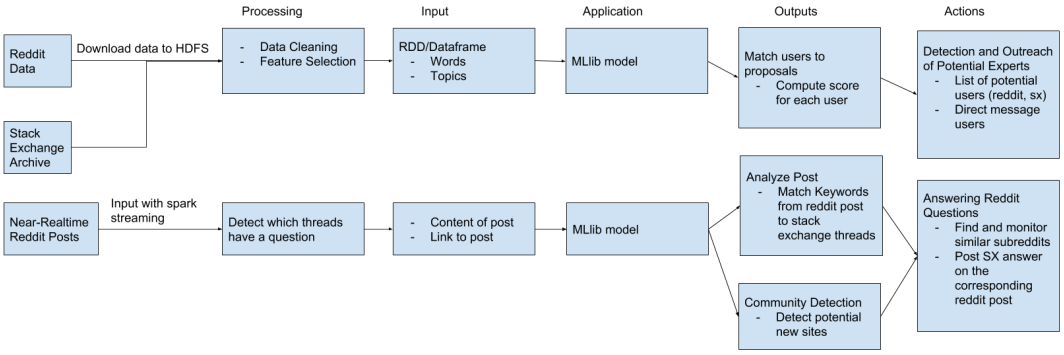

Fig. 1. Project diagram showing the high level flow of the project architecture

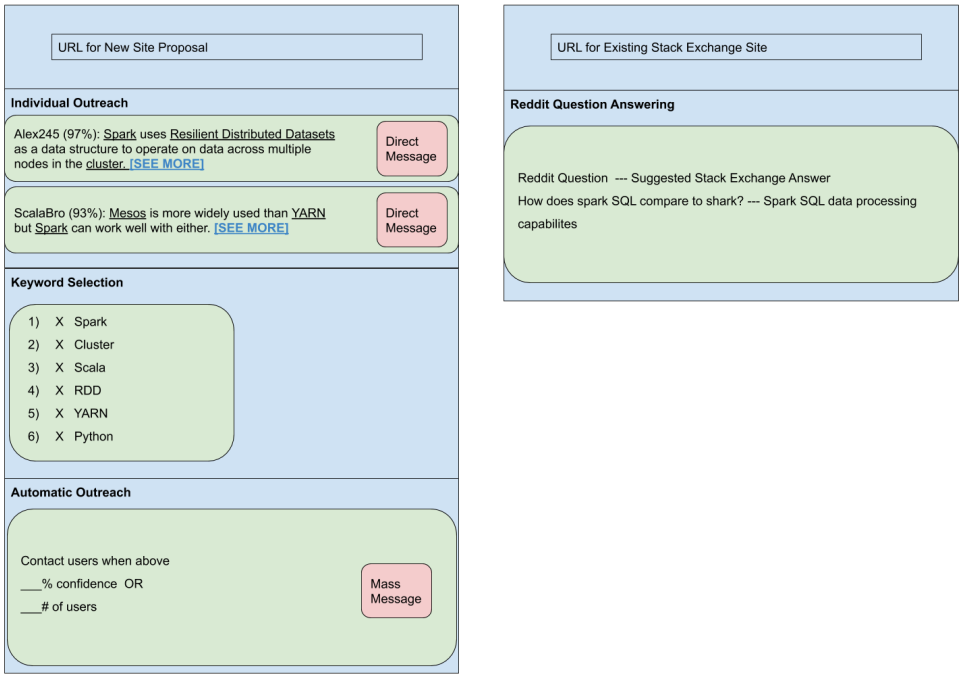

Fig. 2. User interface design

Target Users

This graph shows the most likely users of your new website!

Health Site - Users Analysis

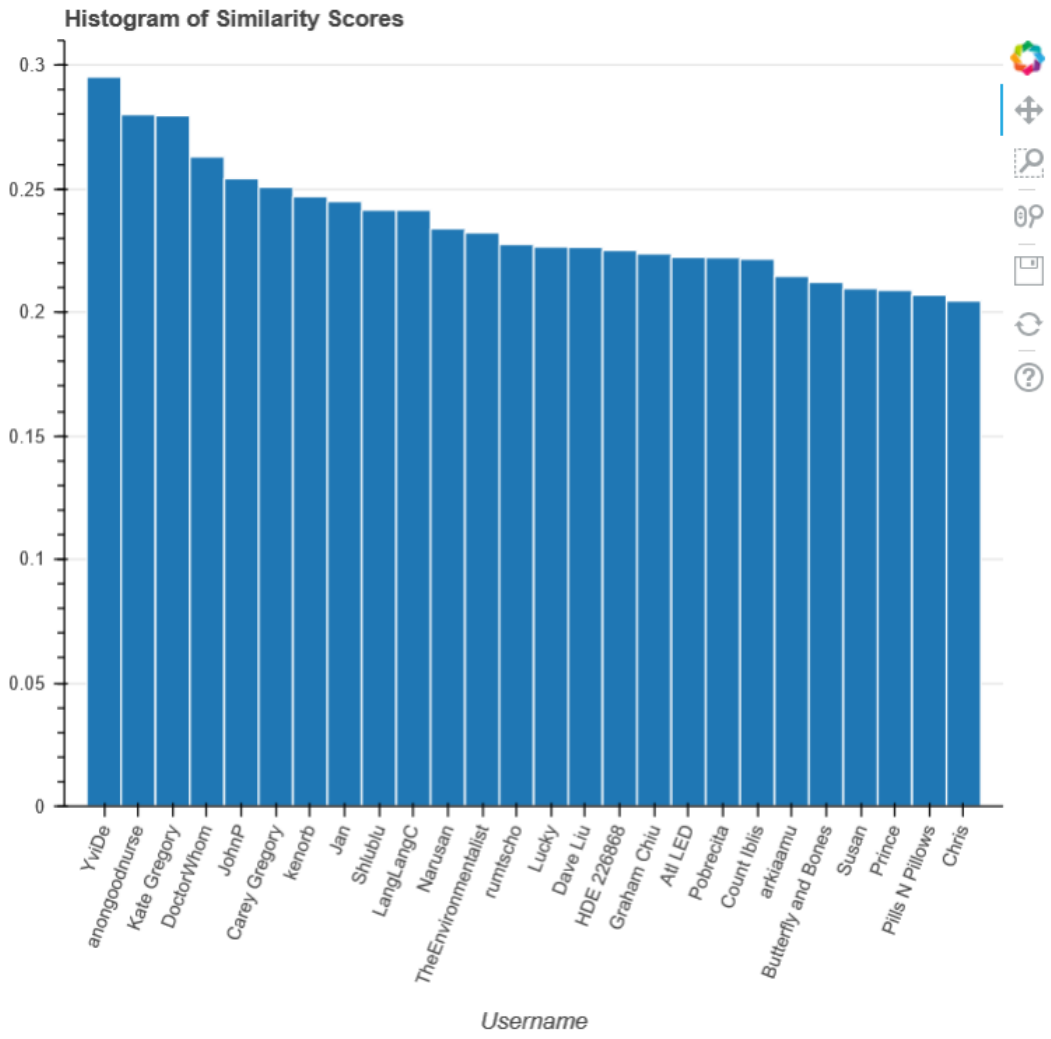

Fig. 3. Similarity scores of users of the Health Stack Exchange

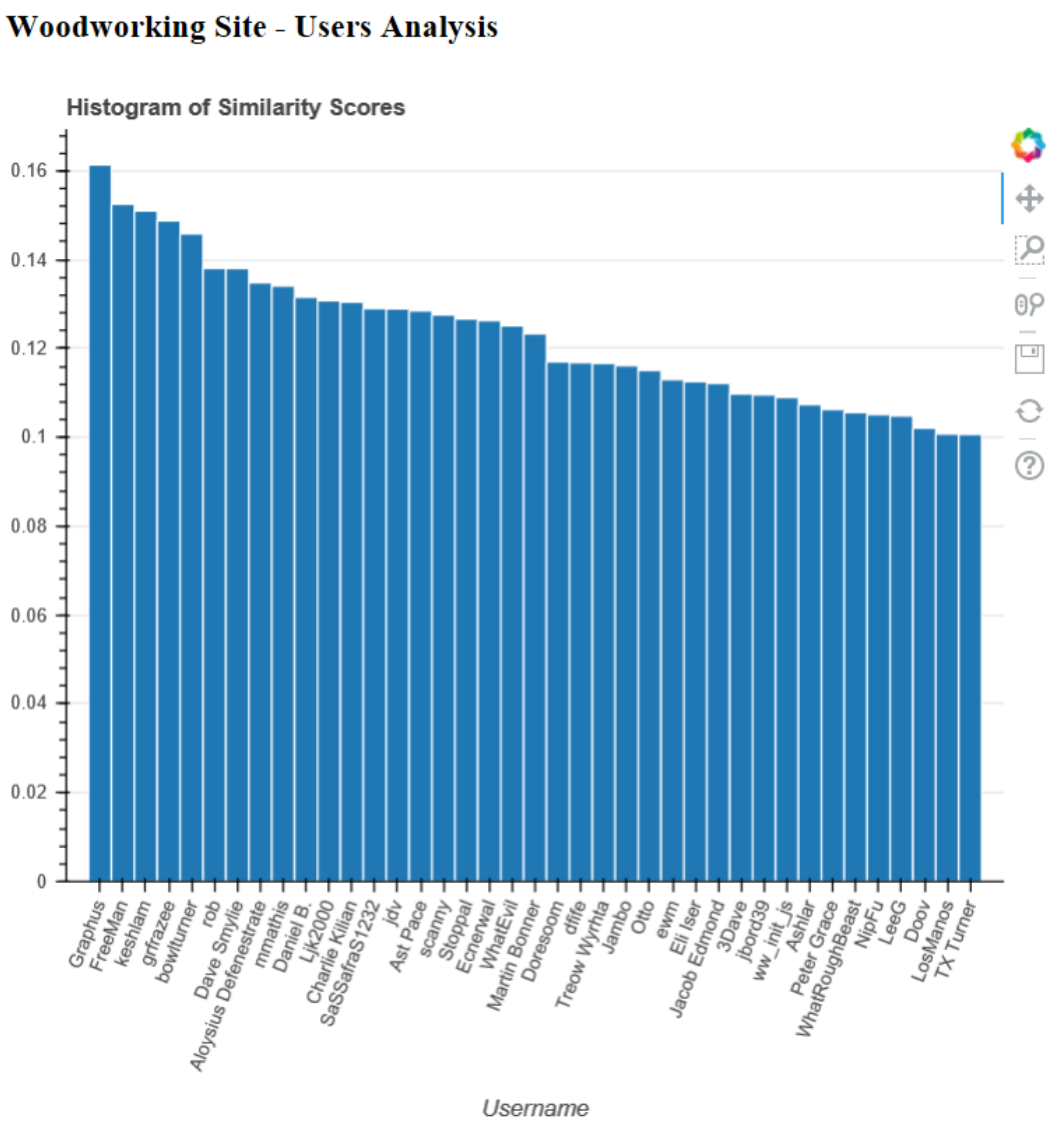

Fig. 4. Similarity scores of users of the Woodworking Stack Exchange

Target Users

This graph shows the most likely users of your new website!

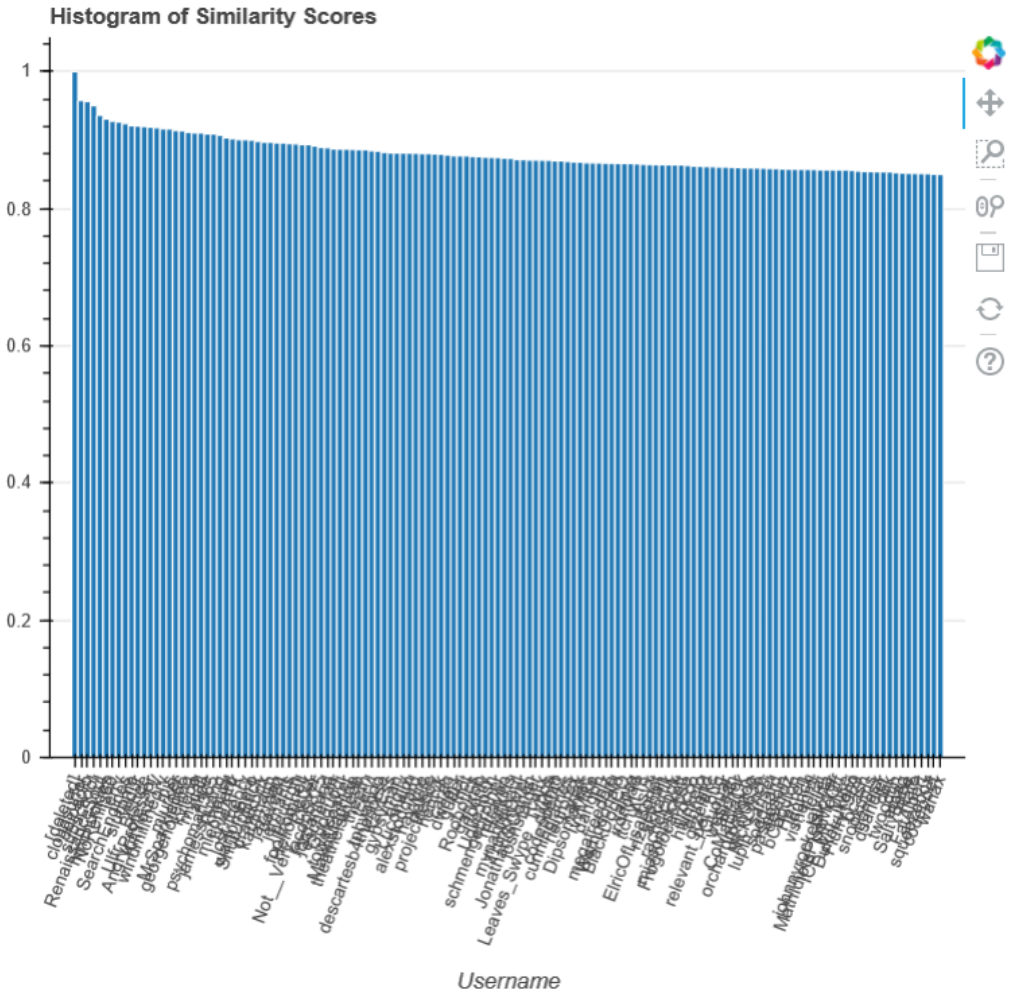

Fig. 5. Similarity scores of Reddit users

Best Candidates for Marketing

Use the list below to select people for marketing

	Link	SimilarityScore
Username		
cloudsurfer	https://www.reddit.com/message/compose/?to=cloudsurfer	0.957815
ddipaolo	https://www.reddit.com/message/compose/?to=ddipaolo	0.956402
subschool	https://www.reddit.com/message/compose/?to=subschool	0.950406
RenaissanceSoul	https://www.reddit.com/message/compose/?to=RenaissanceSoul	0.936201
hmchuckles	https://www.reddit.com/message/compose/?to=hmchuckles	0.930866
NoRefills60	https://www.reddit.com/message/compose/?to=NoRefills60	0.927702
jerry	https://www.reddit.com/message/compose/?to=jerry	0.926477
SearchEngines	https://www.reddit.com/message/compose/?to=SearchEngines	0.924128
slurpme	https://www.reddit.com/message/compose/?to=slurpme	0.921203
vildur	https://www.reddit.com/message/compose/?to=vildur	0.920679
Ulfr__Bishee	https://www.reddit.com/message/compose/?to=Ulfr__Bishee	0.919914
AngryProfessor	https://www.reddit.com/message/compose/?to=AngryProfessor	0.919022
KillYourTV	https://www.reddit.com/message/compose/?to=KillYourTV	0.918186
windmilltheory	https://www.reddit.com/message/compose/?to=windmilltheory	0.916627
mjs	https://www.reddit.com/message/compose/?to=mjs	0.916477
mlurker	https://www.reddit.com/message/compose/?to=mlurker	0.914182
MrSparkle666	https://www.reddit.com/message/compose/?to=MrSparkle666	0.913793
georgehotelling	https://www.reddit.com/message/compose/?to=georgehotelling	0.911575

Fig. 6. Marketing page: allows remediation of sending messages to users of the Reddit based on their likelihood of success