Web Scraping Job Postings for Communication Skills to Target and Improve Engineering Students' Workplace Readiness

Prepared for: Cornell University, College of Engineering
Engineering Communications Program
Alan Zehnder, Associate Dean for Undergraduate Programs

Prepared by: Allison Hutchison, HG Bidon, Aliyah Geer, Angela Liu, Zoe Pan, Amy Qiu, Vishruth Rajinikanth, Ronin Sharma

May 25, 2021

ENGRC 3340 Independent Study Research Team
Cornell University, College of Engineering
Ithaca, NY 14853

CornellEngineering
Engineering Communications Program

Table of Contents

EXECUTIVE SUMMARY

Dr. Allison Hutchison, Senior Lecturer in the Engineering Communications Program, organized this research team along with Drs. Samantha Cosgrove and Bremen Vance after doing a project for several semesters where engineering students analyzed job ads they were interested in applying to. In the short reports that students wrote, Hutchison found initial evidence that communication and collaboration skills were almost always included in engineering job ads, but students often overlooked including these skills in their resumes.

Eight undergraduate CoE researchers sought to understand engineering, risk communication, and UX job ads better by scraping data from Indeed and Glassdoor. The team wanted to answer these three research questions:

- **Research Question 1:** What are the desired communication and collaboration skills mentioned in full-time entry-level jobs and internship advertisements in engineering fields?
- **Research Question 2:** Do trends in communication and/or collaboration skills appear in various job titles?
- **Research Question 3:** How frequently do specific communication and collaboration keywords appear in job advertisements? What does that tell engineering students about the kind of candidates employers are looking for?

Our data collection team selected two popular websites that contain job ads: Indeed and Glassdoor. They collected a total of 5,994 job ads based on 6 different job categories and 22 job titles.

Informed by our study findings, we are suggesting a sound and interactive teaching method that enhances awareness of the importance of technical, communicative, and collaborative skills among all engineering students. In the meantime, instructors should also consciously aim at increasing student's teamwork, communication, writing, speaking, and presentation skills. The team's recommendation of that teaching method is based on the following rationales:

- Communication instruction in engineering programs could increase oral communication, so that the training should not only focus on technical skills, but also the skills that focus on interaction and behavior among their peers.
- The broad category of interpersonal skills that include building and maintaining relationships, using diplomacy, and developing rapport is the utmost expectation

of employees. Thus, engineering students are expected to give constructive criticism and be respectful of others' opinions.

Furthermore, the team suggests more hands-on projects while training engineering students in order to ensure that such skills are instilled in engineering students.

II. INTRODUCTION

Research on engineering alumni highlights the importance of communication and collaboration skills in the workplace [1], [2]. In addition, a 2017 Google study found that its most effective teams possessed power skills, including "good communication, insights about others, and empathetic leadership" [3]. Communication and collaboration are two main learning outcomes in the Engineering Communications Program, and in the ENGRC 3500: Engineering Communications courses taught by Dr. Allison Hutchison, students began to explore these skills in advertisements of jobs they wanted to apply to. In the Spring 2020 semester, Leeds Rising '20, ORIE and CS, suggested that the entire class compare the communication- and collaboration-related skills from the job advertisements they had collected.

What began as an in-class activity has now blossomed into a full research project involving a multidisciplinary team of eight undergraduate students in the College of Engineering, whose majors include ISST, ORIE, CS, and ECE. The students forming this research team enrolled in ENGRC 3340: Independent Study in Engineering Communications in order to take a big data approach to analyzing engineering job advertisements.

A. Background: Technical Communication Skills in Job Advertisements

Technical writing courses have long focused on educating engineers to write effectively [4]. Beginning around the second World War, technical writing began to emerge as its own discipline when "technical writer" became an official job title [5]. A shift began in technical and professional communication (TPC) programs from offering technical writing instruction to engineers to offering programs of study for students to become technical writers. As a result, TPC scholarship has often focused on researching skills and qualifications included in job postings for technical writers [6]–[10]. However, we haven't located any research that examines engineering job advertisements for TPC-related skills. Our research team sought to fill this gap by focusing exclusively on analyzing communication and collaboration skills in engineering job advertisements.

TPC programs are often tasked with educating STEM students through service courses as well as training majors in the program to become future technical writers. Thus, TPC programs are frequently related to workplace readiness and, at times, endeavor to determine whether their courses adequately prepare students for technical communication demands in the workplace. One dominant research method is to conduct surveys of business and engineering alumni [1], [2], [11], [12], technical communication managers and directors of undergraduate TPC programs [6], and job recruiters [10]. Another method is to gather relevant job titles and extract the preferred or required skills from advertisements posted on job search websites [7]–[9]. Our research team took the latter approach.

B. Research Questions

The goal of this research project is to understand engineering job ads better by scraping job listings from employment websites, namely, Indeed and Glassdoor. In particular, the research team looked for desired communication and collaboration skills sought by full-time entry-level and internship job employers in engineering and risk communication fields. Each member of the research team drafted a research question, which we then honed into these three areas of focus:

RQ 1: What are the desired communication and collaboration skills mentioned in full-time entry-level jobs and internship advertisements in engineering fields?

RQ 2: Do trends in communication and/or collaboration skills appear in various job titles?

- Do certain jobs emphasize communication more than others?
- Are skills similar across different dimensions of job ads (i.e. job title, level, company industry)?

RQ 3: How frequently do specific communication and collaboration keywords appear in job advertisements? What does that tell engineering students about the kind of candidates employers are looking for?

C. Summary of Findings

Summary of **RQ 1** findings:

- We totaled the most frequently occurring words that related to communication and collaboration skills: communicate, write, oral, verbal, ask, interpersonal, presentation, team, and collaboration.
- Keywords such as “team,” “collaborate,” and “write” play a significant role in the job advertisements. The occurrence of “team” is the highest, which is a total of 14,639 times out of 5,994 job posts we complied.
- In the overall dataset that includes risk communication, UX, front end, and engineering job advertisements, 67% of those jobs include some form of the word

"communicate." The average number of times the lemma "communicate" is mentioned per job advertisement is 1.63 with a standard deviation of 1.05.
- From the prevalence of "team" in the dataset, we surmise that candidates with strong teamwork skills are highly sought out by employers.

Summary of **RQ 2** findings:

- Every job title in the dataset mentions the above list of communication and collaboration keywords in its advertisements.
- The three jobs with the highest number of mentions of overall communication and collaboration keywords are **Data Engineer**, **Automation Engineer**, and **Technology Analyst**.
- The three jobs with the greatest percentage of mentions of the keyword **communicate** are **Technology Analyst**, **Network Engineer**, and **Data Scientist**.

Summary of **RQ 3** findings:

-

In the rest of the report, the team elaborates on the skills included in various engineering and risk communication job postings.

III. RESEARCH METHODS

We began this research project by reading and discussing selected chapters from Cheryl Geisler and Jason Swarts' book, *Coding Streams of Language: Techniques for the Systematic Coding of Text, Talk, and Other Verbal Data*. While data science and computational linguistics approaches to big data can provide quantitative insights about language as data, this book suggests a mixed methods approach to systematically analyzing patterns in verbal data. In corpus linguistics, for example, data is often analyzed using "grammatical or semantic tagging" [13, p. 5], and in data science, "many big data approaches have little use for interpretation" [13, p. 6]. Because the Engineering Communications Program integrates communication courses with engineering disciplines, we are interested in taking "a rhetorical approach to coding" by "consider[ing] not just what language says [...] but also what language does" [13, p. 11]. Accordingly, we developed a coding scheme according to the process described by Geisler and Swarts in order to hand-code and train the ML model described below.

Based upon the research team's interest in specific engineering jobs, we used X different job titles as keywords to search two commonly used job search websites, Indeed and Glassdoor. The data collection subteam scraped a total of 5,994 jobs from five categories of engineering fields, and the total of advertisements collected for each job title is displayed in Table 1.

TABLE 1
ENGINEERING JOB ADVERTISEMENTS COLLECTED BY TITLE/KEYWORD

Job Title / Keyword	**Job Advertisements**	**Job Category**
Data Analyst	32	Data Engineering
Data Engineer	895	
Data Scientist	425	
Machine Learning Engineer	123	
Network Engineer	314	
Financial Engineer	53	Financial Engineering
Quantitative Analyst	51	
Quantitative Finance	8	
Quantitative Research	10	
Quantitative Trading	49	
Technology Analyst	655	
Automation Engineer	776	Hardware & Technical Engineering
Computer Hardware Engineer	299	
Electrical Engineer	640	
Mechanical Engineer	602	
QA Engineer	152	
Systems Engineer	257	

Software Engineer	653	Software Engineering
Grand Total	5,994	

A. Collecting the Data

In this section, we describe where we collected the data from, what information the data contains, and how we combined all of the data.

1) *Indeed and Glassdoor Web Scraping*

Our data collection team selected two popular websites that contain job ads: Indeed and Glassdoor. We used two Python modules, *requests* and *BeautifulSoup,* to perform the web scraping. Both of these modules are widely used, open-source, and well-documented. First, we created the website that we wanted to obtain data from. To the base Indeed or Glassdoor website, we added the job, job type (internship or full time), and the experience level (entry level). Here is an example Indeed job posting with the search keyword “engineer” and the job type “internship”: https://www.indeed.com/jobs?q=Engineer&jt=internship. We used a similar format for the Glassdoor websites, with the main difference being the website stem, which was Glassdoor instead of Indeed. Next, we used the *get* function from the *requests* module to send a GET HTTPS request to that website. This function call extracted all of the data from that webpage in HTML format. Next, we converted the result of the GET request to a *BeautifulSoup* object, which made it easier to extract relevant information. From the object, we were able to search for specific HTML tags based on attributes such as class or id. This allowed us to find every job on the web page and extract the relevant information from each job.

When scraping from Indeed, we used the *findAll* function from *BeautifulSoup* to extract all the HTML ‘div’ tags. Each div tag contained the data for one job. From this div tag, we extracted the ‘a’ tag with the class attribute ‘jobtitle turnstileLink’. This extracted the title of the job. Similarly, we extracted the company name, salary, location, job description, raw HTML code, and the job URL. The process was very similar for Glassdoor.

In addition to the data from the website, we added some information when saving it to make processing easier. We added a job ID to keep track of jobs, a keyword which was the job we searched for, and whether the job was an internship or entry level position. If any value was missing, we set it to -1. We performed these steps for all 22 job title keywords and collected a total of 8,694 jobs before merging.

2) *Data Merging and Post-Processing*

We scraped job ads from Indeed and Glassdoor separately. We did this because the two websites had completely different website formats, so the raw HTML was different so we couldn't search for the same HTML tags on both websites. As a result, our web scraping results were stored in separate data files. The only modification we had to make during the merge process was some column names. Some column names were slightly different in the two files. For instance, the Indeed data had 'Job Title' as one column while the Glassdoor data had 'Job_Title' as the corresponding column. We used Python's *pandas* module to rename columns and combine the files. Specifically, we loaded in each file as a *pandas* dataframe. We updated the 'columns' attribute of one dataframe to match it to the other dataframe's attribute. During merging we also removed duplicate jobs. We defined duplicates as jobs with the same job title and company name. We searched each dataframe and compared the values at these specific columns. We properly accounted for capitalization when determining duplicates. Note that there were many jobs which had identical job titles and company names, but had different locations. Only one of these jobs were chosen to appear in the final dataset. This prevents the error of the same job with the same description skewing the final results. Finally, we used the *pandas* function 'concat' to add one dataframe below the other. After merging, we were left with 5,994 jobs.

B. Cleaning and Segmenting the Data

In this section, we will describe the data cleaning and segmentation process. We will discuss the tools and methods used to computationally clean and segment our data.

1) *Data Cleaning*

Before segmenting the data, we first had to do some cleaning. The collected data contained columns for the job description text and the HTML version of the description. The job description text was extracted in a way that did not preserve correct separation between some text, so we chose to use the HTML version instead. In order to use this, we had to extract the text from the HTML using *BeautifulSoup*. To preserve the correct separation between text, we added a newline character wherever we removed an HTML tag. We did not do any further cleaning of the text data (i.e. removing punctuation, converting to lowercase) since capitalization and punctuation are useful for segmentation.

2) *Segmentation Tools*

We experimented with using both the *nltk* and *spacy* Python packages to computationally segment our data. Although *nltk* has an easy to use sentence segmenter and performs faster than *spacy*, we settled on using *spacy* since it employs linguistic features in a more sophisticated way. We used *pandas* to manipulate our datasets and write the resulting tables to CSV files.

3) *Sentence Segmentation*

To segment the data into sentences we initially just used *spacy*'s default configurations. The toolkit performs sentence segmentation using information from the dependency parse (a method that encodes relationships between tokens) of the sentence and punctuation. After spot checking the segmented sentences, we noticed that in our data, many strings that we wanted to be segmented as separate sentences were not segmented properly since they were technically incomplete sentences and did not contain punctuation. To solve this, we leveraged the ability to add custom components to the pipeline run by *spacy* when processing text. We added a component to mark tokens that matched a regex sequence as sentence ends. We created a *pandas* table with columns for the id of a sentence, the id of the job whose description it came from, and the text of the sentence.

4) *Noun Phrase Segmentation*

To perform noun phrase segmentation, we used *spacy*'s out-of-the-box support for extracting noun chunks. The *pandas* table for noun chunks contained columns for the noun chunk id, the id of the sentence it appears in, the id of the job it corresponds to, and the noun chunk text.

5) *List Item Segmentation*

We observed in our data that many times the desired skills/qualifications, and job requirements appeared as part of lists. Based on this observation, we decided to also try segmenting the data using <li> tags in the HTML. We used *BeautifulSoup* to find all <li> tags and extract the text from them. The resulting *pandas* table contained columns for the list item id, the id of its corresponding job, and the list item text.

C. Coding the Data

In this section, we will describe the process for coding the segmented data based on the communication categories mentioned. We will talk about the coding scheme we chose as well as our methods for filtering the data and leveraging computational methods to code large volumes of data.

1) *Our Coding Scheme*

We based our coding scheme on the list of communication skills compiled by Dr. Vance and Dr. Cosgrove. They derived this list by first examining 170 articles and labeling the communication skills mentioned in that scholarship, and then following up by surveying employers/hiring managers about those skills. The research team developed a coding scheme by categorizing those communication skills into nine categories (see Table 2 in the Appendix for the full coding scheme).

2) *Filtering the Data*

Since we computationally segmented our data from entire job descriptions, many segments were irrelevant to our research goal. To address this and cut down on unnecessary work of coding irrelevant segments, we filtered the segmented data down to only the segments that contain communication skills from the list by Drs. Vance and Cosgrove.

3) *Coding the Data by Hand*

We began by coding the noun phrases dataset but quickly realized that only using noun phrases excluded many mentions of communication skills in job advertisements. Because of this, we transitioned to coding sentences instead. For a given sentence, we went through each possible communication category and assigned a 0 for a category if it was not mentioned in the segment and a 1 if it was. Although we filtered our data to only contain segments which mentioned communication skills, it was still possible for segments to not belong to any category. For example, many mentions of “write” refer to writing code, and many mentions of “network” are in the context of computer networks.

4) *Leveraging Systematic Coding to Train the Machine Learning Model*

Since we have a very large volume of data (around 27,000 sentences after filtering), we decided to try training a machine learning model to code the data. We used a supervised learning set up, which means we had to first label a portion of our data. We did this by choosing a random subset of 2,000 sentence segments and coding or labeling by hand. We followed the procedures outlined by [13] for systematically applying our coding scheme and coding the segmented data. In that way, we combined both a rhetorical and a data science approach to coding data. Once we hand-coded a subset of our data, we divided it into a training set containing 80% of the data and a test set containing 20%.

In order to train a model on this data, we had to do some preprocessing. This included lemmatizing the segmented text and vectorizing it. To vectorize the text, we used scikit-learn’s TfidfVectorizer. This method creates a vector for each segment that represents the words and/or n-grams (groups of *n* words that appear together) in the segment. It also accounts for how many segments words appear in so that frequently occurring words which are likely stop words (i.e. the, and) receive less weight.

We also noticed that some communication categories had very few examples in the dataset which made it infeasible to train a model to predict those categories. To address this issue, we decided to group the communication categories further, so each group contained more examples. Recall the three categories we developed:

1. **Category 1** included *behavioral* and *interactional* skills;
2. **Category 2** included *aural* and *oral* skills; and
3. **Category 3** included *written* skills, *documentation*, and *style*.

After preprocessing our data, we trained several different types of models including a logistic regression model, a support vector machine, and a multilayer perceptron on the training set and compared their performance using k-fold cross validation. We used k-fold cross validation to tune the model parameters.

To measure model performance, we used F1-score which combines precision and recall into a single metric. Accuracy (percentage of correct predictions) was not a suitable metric for our data since segments that did not belong to a communication category were much more common than those that did. This meant that a model could achieve high accuracy by simply predicting that all segments did not belong to a category. Precision on the other hand measures the percentage of segments the model predicts as belonging to a category that actually belong to that category. Recall measures how well the model does at identifying all the segments belonging to a category. Ideally, we want a model that performs well on both these metrics which is what f1-score captures. We selected the model with the best performance using k-fold cross validation, and then evaluated its performance on the test set.

D. Building the NLP Models

In this section, we will describe the process involved in each step of building the Natural Language Processing Models, and what purpose they will serve to address the aforementioned research questions. With the use of a logistic regression and a Naive Bayes model, we sought to predict whether a job is inherently communicative or not based upon its description. We defined “communicative” in that the given job’s advertising description mentions a need for communication skills. The specific set of skills that we searched for can be found in the coded_filtered_list_items_with_comm_skills sheet.

6) *Pre-Processing the Data*

For the purpose of building the NLP models in such a way that we would have as many descriptive words to work from as possible, our NLP Research Engineer chose to use the non-segmented CSVs from both the Glassdoor and Indeed data collection in order to train our NLP models. With this raw data, we appended an additional column to each of the data sets labelled "Communicative?". This column would return the value 1 if **any** of the 89 phrases found in the aforementioned CSV sheet was found in the corresponding job description. If not, the column would be inputted with the value 0. This "Communicative?" column would serve as the response variable for the NLP models to be characterized later in this section. After dropping irrelevant columns such as "Company," "Location," and "Salary," the two data frames were combined into a single data frame containing each of 6,036 job descriptions along with their respective keyword and whether the job description requires communication skills.

7) *Text Processing*

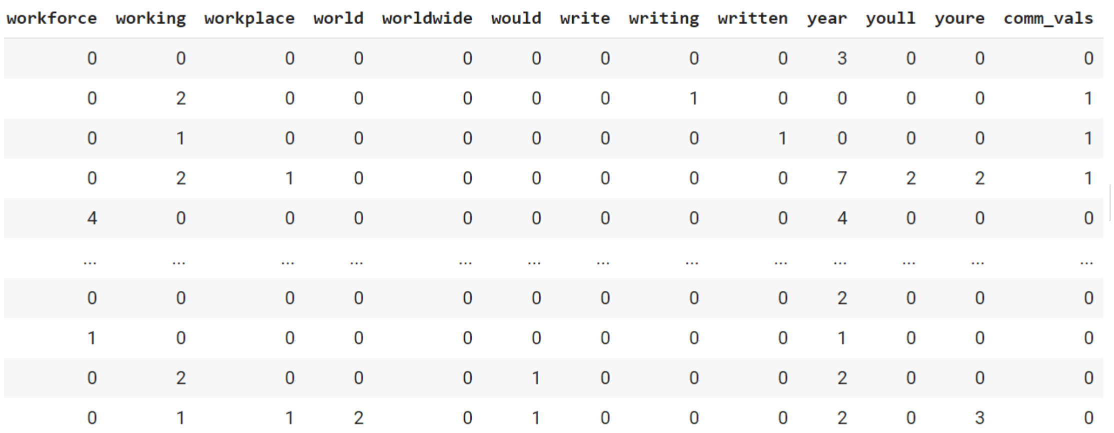
For the ease of count vectorization (to be discussed more in detail in the next section), we cleaned much of the description portion of the data by eliminating stopwords, setting it all to lowercase, and removing punctuation. Stopwords can be defined as words that do not add to the meaning of a piece of text, or "unimportant" words such as: this, and, are, is, and so on. Although removing punctuation would inhibit noun/verb segmentation of the descriptions, this proved to be inconsequential to this particular analysis as we decided to move forward without segmentation here.

8) *Count Vectorization*

At this point in the process, we performed feature extraction so that our model can readily analyze the data. This means that we converted each row of the data set's description section into a vector of up to 1000 words using Sci-kit Learn's CountVectorizer function. Each column of the resulting matrix represented a different word, with each entry a count of how many times that word appeared in the row's job description. The matrix was then appended with the "Communicative?" column mentioned earlier. A partial image of the vectorized matrix is in Table 3 below.

TABLE 3

COUNT VECTORIZATION



workforce	working	workplace	world	worldwide	would	write	writing	written	year	youll	youre	comm_vals
0	0	0	0	0	0	0	0	0	3	0	0	0
0	2	0	0	0	0	0	1	0	0	0	0	1
0	1	0	0	0	0	0	0	1	0	0	0	1
0	2	1	0	0	0	0	0	0	7	2	2	1
4	0	0	0	0	0	0	0	0	4	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...
0	0	0	0	0	0	0	0	0	2	0	0	0
1	0	0	0	0	0	0	0	0	1	0	0	0
0	2	0	0	0	1	0	0	0	2	0	0	0
0	1	1	2	0	1	0	0	0	2	0	3	0

Each entry of the count vectorization matrix represents a count of the column-specific word for each of the job listings, with each column being a word present in the job descriptions. The far right column, “comm_vals”, is zero if the job has been labelled non-communicative, and one if it has been labelled communicative.

9) *The NLP Models*

We separated the data into a training set and a test set, with 25% of the data randomly split into a test set and the rest into a training set. The first model we chose to use here was a naive Bayes, which is a model that classifies data using the label with the highest conditional probability. This model is called “naive” because it makes the strong assumption that all variables are independent, which is not usually the case. However, it usually is a good binary classifier to start a project given its simplicity. The next model we chose to use for this project is a logistic regression, which is historically an excellent model to use for binary classification. Essentially, it fits a logistic function to the data in order to maximize that the observations follow the curve. Many NLP Engineers use this model due to how easily its findings can be interpreted into probabilities. However, it does follow a certain set of assumptions: that there is a linear relationship between the predictors and the log-odds of the response variable, that observations are independent of each other, and that the data has low multicollinearity. Although it would be its own time-consuming task to analyze whether these assumptions are true, we can still use the logistic regression model, and reevaluate if need be.

After completing the general analysis of predicting whether any particular advertisement can be described as communicative, we sought to look into specific job titles to find a correlation between the keyword associated with a job and the level of communication required from that respective description. Thus, we separated the datasets by keyword and evaluated each model performance on a keyword-by-keyword basis.

10) Evaluation

We can evaluate the model's performance by examining three key metrics: precision, recall, and F1-Score. Precision can be defined as the number of "true positive" values over the number of predicted positive values. It measures how precise the model is when it predicts a positive value, or in this case, a 1 in the "Communicative?" column. Recall, on the other hand, can be defined as the number of true positive values over the number of actual positive values. Finally, the F1- Score is a weighted combination of the former two metrics. The exact formula is below:

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Additionally, we will want to pay attention to how high the training metrics are versus the testing metrics. A training accuracy that is very high while paired with a very low testing accuracy is indicative of overfitting, a common problem in machine learning. A testing accuracy above 50% is usually considered "good" in the case of binary classification because that would indicate that the model is simply not randomly guessing a prediction. Anything much higher than 70% would traditionally be considered very good, although it depends on the application.

IV. RESULTS

Here we report general findings from the overall dataset as well as more specific findings from the two models.

A. Desired Communication and Collaboration Skills in Engineering Fields

As mentioned above, hand-coding even the filtered dataset—the one with communication and collaboration keywords—would be arduous with roughly 27,000 rows segmented by sentences. Therefore, we implemented an automated coding formula in Excel [13, p. 146] to quantify the occurrence of communication and collaboration keywords in the dataset. We selected specific keywords from the list of communication skills provided by Drs. Cosgrove and Vance, and we corroborated this selection with keywords that we noticed frequently while coding the 2,000-sentence segment sample of data.

RQ 1: *What are the desired communication and collaboration skills mentioned in full-time or entry-level jobs or internship advertisements in engineering fields?*

We sought to answer the first research question by totaling the most frequently occurring words that related to communication and collaboration skills: **communicate, write**, **oral**, **verbal**, **ask**, **interpersonal**, **presentation**, **team**, and **collaboration**. These words were totaled using a wildcard function, meaning that any word with the same base before the asterisk would be counted in the dataset in an attempt to lemmatise the keywords.

By far, the most frequently occurring lemma was **team**, returning 14,639 instances. The second most frequently-used lemma was **communicate** with 6,546 occurrences. **Write** followed in third with 4,186 mentions. **Oral** and **verbal** skills were also frequently mentioned, with a combined total of 2,617 occurrences. We also find it noteworthy that **presentation** is mentioned 1,035 times.

In the overall dataset that includes risk communication, UX, front end, and engineering job advertisements, 67% of those jobs mention **communicate**. The average number of times the lemma **communicate** is mentioned per job advertisement is 1.63 with a standard deviation of 1.05.

In addition to looking at the number of occurrences of certain words, we also analysed what words were most important in classifying a sentence into one of three categories defined as follows:

- Category 1: Behavioral and Interactional
- Category 2: Aural and Oral
- Category 3: Written, Documentation, and Style

We did this by using the weights from the trained NLP model as a score. A higher weight corresponds to a higher score because it is a “more important” factor. The scores were then used to generate word clouds for the three categories shown in Figures 1, 2, and 3.

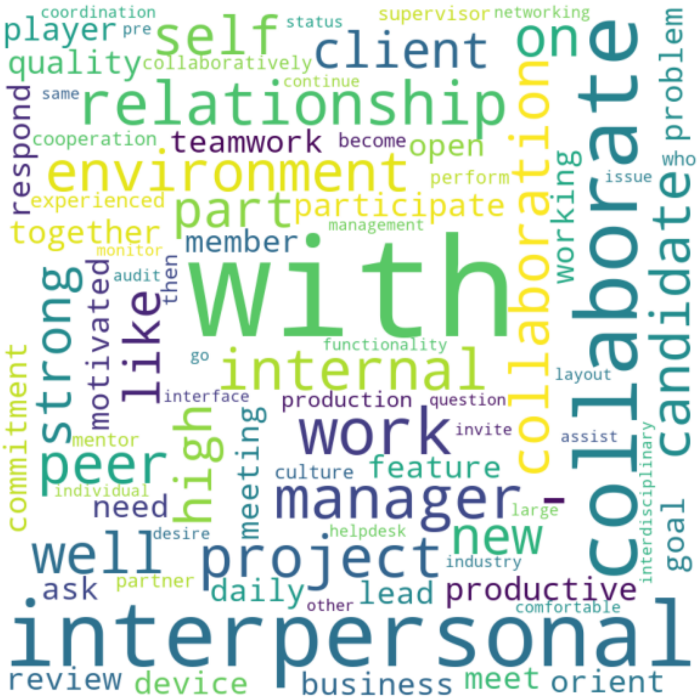


Fig. 1. Word Cloud For Category 1: Behavioral and Interactional

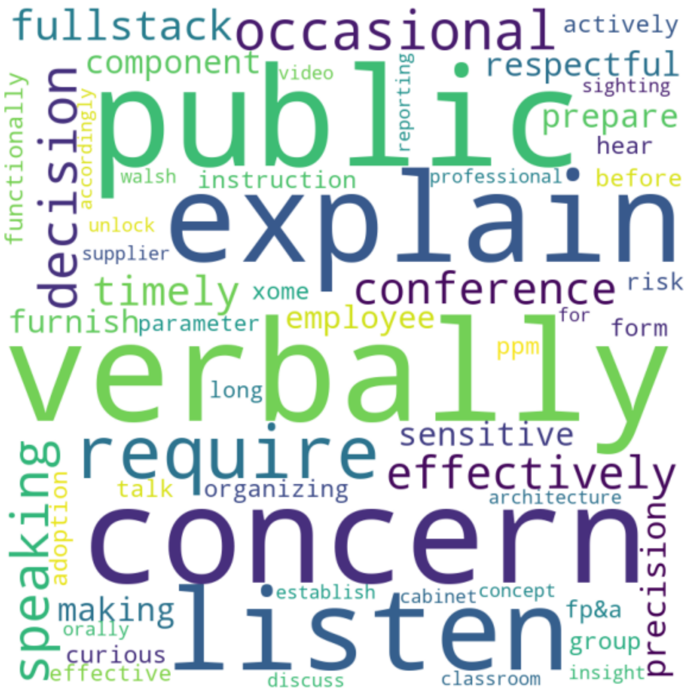


Fig. 2. Word Cloud For Category 2: Aural and Oral

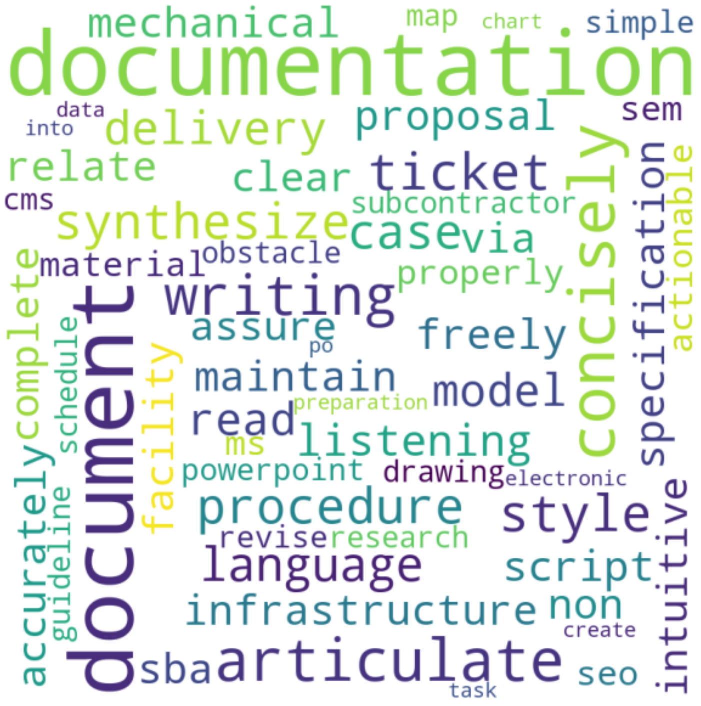


Fig. 3. Word Cloud For Category 3: Written, Documentation, and Style

B. Trends in Communication and Collaboration Skills by Engineering Job Title

To address the second research question, we created pivot tables—from the same set of automated coding spreadsheet mentioned above—of the specific communication and collaboration keywords by job title. A total for each of these keywords broken down by job title is provided in Table 4 (see Appendix).

RQ 2: *Do trends in communication and/or collaboration skills appear in various job fields?*

- *Do certain jobs emphasize communication more than others?*
- *Are skills similar across different dimensions of job ads (i.e. job title, level, company industry)?*

By combining the total number of occurrences of communication and collaboration keywords by job title, the three jobs with the highest mentions were:

1. **Data Engineer:** 5,618 mentions
2. **Automation Engineer:** 4,895 mentions
3. **Technology Analyst:** 4,735 mentions

Software Engineer is also noteworthy at 3,938 mentions across all the advertisements with that title. While these four jobs top the list, every job title in the dataset mentions the above list of communication and collaboration keywords in its advertisements.

Focusing on mentions of **communicate**, the dataset reveals a slightly different pattern that indicates an answer to the bulleted sub-questions. In Table 5 (see Appendix), each job advertisement that mentions the keyword **communicate** was counted once and totaled by job title. That number was then divided by the total number of advertisements collected with that job title. By this metric, the three jobs with the greatest percentage of mentions of **communicate** are:

1. **Technology Analyst:** 76%
2. **Network Engineer:** 74%
3. **Data Scientist:** 71%

While **Technology Analyst** again appears in the top three, **Network Engineer** and **Data Scientist** emerged as job titles with a great deal of the keyword **communicate** in their advertisements.

B. Results from the ML Model

The final ML models, which were trained by segmenting the data into n-grams, were able to classify sentences into Category 1, Category 2, and Category 3 with the following presion, recall, and F1 scores shown in Table 6.

TABLE 6
TESTING ACCURACIES FOR SELECTED MODELS

	Model Selected	Precision	Recall	F1 Score
Category 1: Behavioral and Interactional	Logistic regression using lemmatized unigrams	0.65	0.56	0.60
Category 2: Aural and Oral	SVM linear kernel using non-lemmatized unigrams	0.71	0.69	0.70
Category 3: Written, Documentation, and Style	SVM linear kernel using lemmatized unigrams	0.73	0.68	0.71

The lower F1 score for Category 1 correlates with a lower simple agreement of inter-rater reliability score for the interactional label. In particular, using a random subset of 130 rows, the inter-rater reliability for the interactional column was 69.23%.

In addition to using the ML models to computationally code our segmented data, we also performed some data analysis using what our models learned. Linear ML models learn a vector of weights with indices corresponding to vector indices in the input. Since we vectorized our data so that each index of a vector corresponds to a word or n-gram, this means each weight learned by the model corresponds to a word or n-gram. Words that receive high positive weights in the model for a category are the words that are most representative of that category. Words with low negative weights are the words that are least representative of a category.

C. Results from the NLP Models

Table 7 below depicts the testing classification metrics from the naive Bayes model for the general analysis. With a weighted average of 70% testing accuracy, we can consider this to be, in general, a fairly good model for our purposes. This means that the model can accurately predict whether a job is "communicative" 70% of the time.

TABLE 7

TESTING ACCURACIES FOR NAIVE BAYES

	Precision	Recall	F1 Score
0 (Non-Communicative)	.77	.72	.74
1 (Communicative)	.60	.65	.63
Weighted Average	.70	.70	.70

Table 8 below shows the testing classification metrics for the logistic regression model with the general data set. The logistic regression tends to perform slightly better than the Naive Bayes on the whole, likely because the logistic regression model was formulated specifically for binary classification, while the Naive Bayes model is a simple probabilistic model that assumes strong feature independence, which may not be the case with our data. With these conclusions in mind, we can better understand the results from the job-specific models.

TABLE 8

TESTING ACCURACIES FOR LOGISTIC REGRESSION

	Precision	Recall	F1 Score
0 (Non-Communicative)	.85	.87	.86
1 (Communicative)	.79	.76	.78
Weighted Average	.83	.83	.83

For the purposes of this project, our team looked exclusively at the weighted average of the F1-scores to evaluate model performance. These values are listed in Table 9 for each model and their associated job title keyword. The yellow highlighted entries represent "stand-out" model performances, with at least 75% average accuracy across both models. On the other hand, the light red highlighted entries account for the particularly poor performing keywords.

TABLE 9
WEIGHTED TESTING ACCURACIES BY KEYWORD

Job Keyword	Weighted Average (Logistic Regression)	Weighted Average (Naive Bayes)	Combined Model Average
Automation Engineer	.81	.71	0.76
Computer Hardware Engineer	.68	.60	0.64
Data Analyst	.63	.57	0.6
Data Engineer	.78	.73	0.755
Data Scientist	.62	.60	0.61
Electrical Engineer	.72	.73	0.725
Financial Engineer	.55	.60	0.575
Machine Learning Engineer	.66	.70	0.68
Mechanical Engineer	.72	.70	0.71
Network Engineer	.70	.71	0.705
QA Engineer	.56	.66	0.61
Quantitative Analyst	.57	.71	0.64
Quantitative Trading	.62	.47	0.545
Software Engineer	.78	.65	0.715
Systems Engineer	.80	.76	0.78
Technology Analyst	.65	.68	0.665

From these results, we conclude that jobs falling under the keywords of **Automation Engineer**, **Systems Engineer**, and **Data Engineer** are more easily identifiable as communicative or non-communicative, while **Financial Engineer** and

Quantitative Trading are far more ambiguous. This could be because these jobs may contain words that signalled a communicative job to the model, but truly were not (false positive). Or, vice versa, where the job description did not contain any words that signalled a communicative job to the model, but did in fact end up being classified as communicative (false negative). Importantly, the model was not trained with the 89 listed communication words (see Table 2 in the Appendix), so it did not necessarily look for these words when classifying each job. However, this also could be the result of there being far fewer job advertisements with those titles in our dataset. These results are semi-inconclusive as they show no clear indication towards a particular job keyword being more or less communicative than others, which does not help to answer Research Question 2: *Do trends in communication and/or collaboration skills appear in various job fields?* In the future, this issue could be amended by tailoring the research questions more closely to our intended methods.

V. DISCUSSION

We began with a rather ambitious project to analyze a large dataset of nearly 6,000 engineering job advertisements. Above all else, as lead researcher on this project, Dr. Hutchison acknowledges the undergraduate research team for their extraordinary dedication, teamwork, and persistence during the challenging Spring 2021 semester. We worked remotely, meeting in two groups twice per week, in the midst of the ongoing pandemic, increases in hatred and violence toward Asian Americans and Pacific Islanders, multiple police shootings, and even deaths within the Cornell student community. While we all began this project with high hopes and efforts, our resilience was unmatched to the tragedies that this semester brought to each of us. It is no small feat that we accomplished what is described in this recommendation report.

A. Shortcomings of this Research

For the results in which an automated coding formula was applied to the dataset, not all keyword mentions can be considered valid according to our coding scheme. As we explain in Coding the Data by Hand, some communication and collaboration keywords would be excluded by the rhetorical coding approach because they aren't a reference to skills or qualifications. For instance, this **communicate** reference, "Network communication via CAN per SAE J1939," is irrelevant to our current research questions because it does not refer to human communication. Another example is this **write** reference, "Experience writing testable code and shipping code into production," because it refers to writing code rather than documents or emails. Therefore, as this research project moves forward, the coding scheme needs to be refined in order to calculate inter-rater reliability. So far, we have calculated simple agreement on a subsample of the 2,000-segment dataset. Two of the codes, **multimodal** and **oral**, had rather simple high agreement at 90% and 85.38%, respectively. However, **interactional** had lower agreement at 69.23%, again indicating that the coding scheme needs further refinement.

VI. RECOMMENDATIONS

In our study, our researchers wanted to understand the importance of communication and collaboration skills expressed in risk communication, UX, and engineering job advertisements. We also attempted to detect the trends in communication and collaboration skills for various engineering jobs. After identifying the nature of the industry mindset, we turned our attention to recognize the connection between the company's culture and the candidates' traits. Underpinned by our study findings, we recommend a sound and interactive teaching method that enhances awareness of the importance of technical, communicative, and collaborative skills among engineering students. Instructors should consciously aim at increasing student's teamwork, communication, writing, speaking, and presentation skills.

We believe that communication instruction in engineering programs could increase oral communication, including the skills required for successful interpersonal and teamwork interactions, and should not only focus on technical skills. Most engineering jobs such as Technology Analyst, Network Engineer, and Data Scientist include communication skills in their advertisements.

Apart from sharpening oral and writing skills, candidates should improve their interaction and behavior among their peers; this will enable them to quickly get along with their future workmates when in the workplace. Employees are always part of a team. Employers will always want employees, even those not in an official team, to collaborate with their mates. While one may prefer to work alone, the demonstration that an individual understands and appreciates the value of working in partnerships and joining forces in accomplishing the goals of an organization is essential in workplace setup. The broad category of interpersonal skills that include building and maintaining relationships, using diplomacy, and developing rapport is the utmost expectation of employees. Thus, students should be able to equip themselves with them in readiness for employment. As a broader skill required in teamwork, engineers are expected to give constructive criticism and be tolerant and respectful of their workmates' opinions. Respect and tolerance are two crucial behavioral skills central to building a solid foundation of accountability and trust.

We further recommend the introduction of more hands-on projects that combine theoretical engineering knowledge and incorporate cross-cultural communication. The study found out that companies tend to employ or engage people who share in their culture. Students need to experience cultural variability by interacting with the websites of different companies or interacting with workers/engineers in their fields. Many companies do not like micromanaging their employees. They expect them to be very responsible and do their work without being supervised. Managers demand punctuality, meeting deadlines, and delivering error-free products from their team members. Commitment and excellent performance are some of the practices that companies have accepted to be part of their cultures. Organizations prefer having employees that can easily conform to their cultures. Hands-on projects will ensure that such skills are instilled in engineering students.

Many companies and organizations consider communication and collaboration skills important for engineering jobs. Students who have already developed and improved their interaction and behavioral elements are advantaged. However, for a holistic preparation of students to be ready for the workplace, the already packed engineering curriculum should incorporate additional competency-based courses such as interaction and intercultural skills, especially in communication. The integration of communication skills will serve to prepare engineering students for job openings.

REFERENCES

[1] P. Sageev and C. J. Romanowski, “A Message from Recent Engineering Graduates in the Workplace: Results of a Survey on Technical Communication Skills,” *J. Eng. Educ.*, vol. 90, no. 4, pp. 685–693, 2001, doi: 10.1002/j.2168-9830.2001.tb00660.x.

[2] D. Cunningham and J. Stewart, “Perceptions and Practices: A Survey of Professional Engineers and Architects,” *ISRN Education*, 2012. https://www.hindawi.com/journals/isrn/2012/617137/ (accessed Mar. 11, 2020).

[3] A. Agarwal, “Data Reveals Why the ‘Soft’ In ‘Soft Skills’ Is A Major Misnomer,” *Forbes*. https://www.forbes.com/sites/anantagarwal/2018/10/02/data-reveals-why-the-soft-in-soft-skills-is-a-major-misnomer/ (accessed Mar. 11, 2020).

[4] R. J. Connors, “The Rise of Technical Writing Instruction in America,” *J. Tech. Writ. Commun.*, vol. 12, no. 4, pp. 1–1, Jan. 1983, doi: 10.2190/793K-X49Q-XG7M-C1ED.

[5] F. M. O’Hara, “A Brief History of Technical Communication,” in *Annual Conference of the Society for Technical Communication*, 2001, vol. 48, pp. 500–504.

[6] K. T. Rainey, R. K. Turner, and D. Dayton, “Do curricula correspond to managerial expectations? Core competencies for technical communicators,” *Tech. Commun.*, vol. 52, no. 3, pp. 323–352, 2005.

[7] C. Lauer and E. Brumberger, “Technical Communication as User Experience in a Broadening Industry Landscape,” *Tech. Commun.*, vol. 63, no. 3, pp. 248–264, 2016.

[8] E. Brumberger and C. Lauer, “The Evolution of Technical Communication: An Analysis of Industry Job Postings,” *Tech. Commun.*, vol. 62, no. 4, pp. 224–243, 2015.

[9] C. R. Lanier, “Analysis of the Skills Called for by Technical Communication Employers in Recruitment Postings,” *Tech. Commun.*, vol. 56, no. 1, pp. 51–61, 2009.

[10] R. Stanton, “Do Technical/Professional Writing (TPW) Programs Offer What Students Need for Their Start in the Workplace? A Comparison of Requirements in Program Curricula and Job Ads in Industry,” *Tech. Commun.*, vol. 64, no. 3, pp. 223–236, 2017.

[11] J. M. Huegli and H. D. Tschirgi, “An Investigation of Communication Skills Application and Effectiveness At the Entry Job Level,” *J. Bus. Commun. 1973*, vol. 12, no. 1, pp. 24–29, Oct. 1974, doi: 10.1177/002194367401200104.

[12] A. L. Darling and D. P. Dannels, “Practicing Engineers Talk about the Importance of Talk: A Report on the Role of Oral Communication in the Workplace,” *Commun. Educ.*, vol. 52, no. 1, pp. 1–16, Jan. 2003, doi: 10.1080/03634520302457.

[13] C. Geisler and J. Swarts, *Coding Streams of Language: Techniques for the Systematic Coding of Text, Talk, and Other Verbal Data*. WAC Clearinghouse, 2019.

APPENDIX

TABLE 2
Coding Scheme

List of Communication Skills[1]	**Code (Label)**	**Coding Description**
Listening skills	Aural	receiving verbal or nonverbal information through hearing
Silence	Aural	
Demonstrate respect	Behavioral	describes desired behavior when communicating, often physical or personal such as a character trait or "professionalism"
Courtesy	Behavioral	
Politeness	Behavioral	
Approachable	Behavioral	
Responsiveness	Behavioral	
Eye contact	Behavioral	
Express sympathy/empathy	Behavioral	
Tact	Behavioral	
Express gratitude	Behavioral	
Facial expressions	Behavioral	
Maintain composure in front of audience	Behavioral	
Build rapport	Behavioral	
Convey confidence	Behavioral	
Body position	Behavioral	
Interpersonal communication skills	Interactional	requires interaction with one or more coworkers, clients, or other stakeholders
Address others	Interactional	
Answer questions	Interactional	

[1] We thank Drs. Samantha Cosgrove and Bremen Vance, our research partners, for providing this list based upon their research on qualifications from technical and professional communication jobs.

Engage in conversation	Interactional	
Participates in group discussion/meetings	Interactional	
Ask for opinions	Interactional	
Contribute to meetings/teams	Interactional	
Intergenerational communication	Interactional	
Make introductions	Interactional	
Network with colleagues	Interactional	
Give feedback (positive and negative)	Interactional	
Incorporate feedback	Interactional	
Give instructions (to colleagues and/or subordinates)	Interactional	
Create small talk	Interactional	
Conversation management	Interactional	
Cross-cultural/intercultural communication (verbal and nonverbal)	Interactional	
Negotiate	Interactional	
Avoid discriminating language	Interactional	
Ask questions	Interactional	
Convey interest to clients	Interactional	
Express encouragement	Interactional	
Leadership communication skills	Interactional	
Face-to-face communication	Multimodal	any communication that requires more than one mode simultaneously or during a specific event/interaction (ex: telephone skills require oral and aural skills, presentations require oral and visual skills)
Electronic communication	Multimodal	
Nonverbal communication	Multimodal	
Telephone skills	Multimodal	

Use multiple modes of communication	Multimodal	or can take place in various modes
Create presentations	Multimodal	
Paraphrase	Multimodal	
Oral communication skills	Oral	describes communication based upon spoken language and/or skills
Pronunciation	Oral	
Enunciation	Oral	
Deliver oral presentations	Oral	
Inflection	Oral	
Public speaking	Oral	
Impromptu speaking	Oral	
Give an "elevator" speech	Oral	
Accurate messages	Rhetorical	refers to the audience, context, or purpose of communication; description often begins with a verb
Correct messages	Rhetorical	
Adapt to the situation/audience	Rhetorical	
Articulate purpose	Rhetorical	
Persuasive speaking & writing	Rhetorical	
Articulate ideas	Rhetorical	
Clarify information	Rhetorical	
Explain	Rhetorical	
Give examples	Rhetorical	
Make requests	Rhetorical	
Communicate bad news messages	Rhetorical	
Coherent messages	Style	describes how the communication should be
Complete messages	Style	

Clarity: messages that are explicit, simple, and compact	Style	standardized or formatted (ex: clarity, concision, coherence)
Organized messages	Style	
Tone	Style	
Concise messages	Style	
Concrete messages	Style	
Transition between ideas	Style	
Grammar skills	Written	refers to any text-based or linguistic communication
Email	Written	
Spelling	Written	
Writing	Written	
Punctuation	Written	
Sentence structure	Written	
Proofread	Written	
Active voice/verb usage	Written	
Write questions	Written	
Draft (writing)	Written	
Write memos and letters	Written	
Write formal and informal letters	Written	
Write instructions	Written	
Write reports	Written	
Paragraphing	Written	
Write/prepare a speech	Written	
Write code and/or documentation	Documentation	specifically refers to reading or writing code documentation

TABLE 4

TOTAL NUMBER OF COMMUNICATION AND COLLABORATION KEYWORDS PER JOB

Job Title	communicat*	writ*	oral*	verbal*	ask*	interpersonal	presentation	team*	collaborat*
Automation Engineer	821	579	128	221	419	110	112	2013	492
Computer Hardware Engineer	405	229	46	84	152	61	35	753	164
Data Analyst	30	18	2	5	9	5	9	62	14
Data Engineer	950	650	124	243	458	113	164	2259	657
Data Scientist	448	283	88	103	201	56	121	1166	363
Electrical Engineer	686	406	121	192	361	95	105	1064	208
Financial Engineer	39	49	10	14	34	8	10	141	30
Machine Learning Engineer	94	57	14	28	50	14	19	326	78
Mechanical Engineer	551	406	102	188	309	102	98	1218	209
Network Engineer	455	184	47	88	153	46	42	742	180
QA Engineer	154	106	19	48	147	23	22	331	60
Quantitative Analyst	57	38	9	13	17	12	18	108	37
Quantitative Finance	7	4	0	2	2	0	3	29	9
Quantitative Research	17	9	1	2	5	0	5	34	13

Quantitative Trading	36	33	6	12	25	6	7	101	30
Software Engineer	588	429	70	155	284	63	60	1880	409
Systems Engineer	246	168	45	59	142	18	35	620	129
Technology Analyst	962	538	112	216	419	156	170	1792	370
Total mentions by lemma	**6546**	**4186**	**944**	**1673**	**3187**	**888**	**1035**	**14639**	**3452**

TABLE 5

COUNT OF “COMMUNICATE” IN ADVERTISEMENTS BY JOB TITLE

Job Title/Keyword	No. of Job Ads Mentioning Communication	Total No. of Job Ads Collected	% of Job Ads Mentioning Communication
Automation Engineer	515	776	66%
Computer Hardware Engineer	206	299	69%
Data Analyst	18	32	56%
Data Engineer	600	895	67%
Data Scientist	300	425	71%
Electrical Engineer	437	640	68%
Financial Engineer	28	53	53%
Machine Learning Engineer	71	123	58%
Mechanical Engineer	376	602	62%
Network Engineer	231	314	74%
QA Engineer	94	152	62%
Quantitative Analyst	42	51	82%
Quantitative Finance	5	8	63%
Quantitative Research	8	10	80%
Quantitative Trading	31	49	63%
Software Engineer	390	653	60%
Systems Engineer	163	257	63%
Technology Analyst	500	655	76%

Each job advertisement that mentioned the lemma “communicate” was counted once and totaled here. While some advertisements mentioned “communicate” multiple times, each was only counted once. Job titles highlighted in yellow had the highest percentage of mentions from the total number of advertisements collected. Job titles highlighted in light blue have higher percentages but are less representative of the full dataset due to having a fewer number of total advertisements collected.