

# Cornell Daily Sun Age to Article Classification

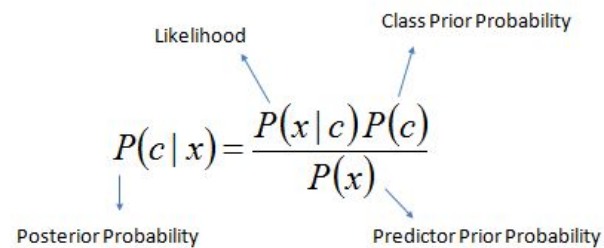
By Mindy Lou (mml234) and Chris Sciavolino (cds253)

## Goals

The goal of our project is to provide a machine learning model that can successfully classify the age group most likely to read a particular article from the Cornell Daily Sun. For training data, we used information obtained from Google Analytics on the Cornell Daily Sun website. This includes various information about users such as articles read, age, common categories, and other useful data that could be used as features for our model. The Google Analytics store goes back over a year, so the data is recent, plentiful, and reflective of the current population of Cornell undergraduates as well as other readers, such as alumni, faculty, and the local Ithaca community. We grouped our ages into three buckets: 18 - 24, 25 - 44, and 45+. This is meant to represent younger individuals and undergraduate students (18 - 24), recent graduates and those early in their careers (25 - 44), and older alumni and readers (45+). We plan to demonstrate the abilities of our Naive Bayesian classifier in the form of an iOS prototype application that will match the latest news articles from the Cornell Sun to the age group that would be most likely to read them. To reiterate, the goal of our model is to take in an article from the Cornell Daily Sun and predict the age group of the most likely reader. The information gained from this project could potentially assist the Cornell Sun with their article strategy and how they plan to target current and future readers with the content that they are producing, and improve their current article recommendation system.

## Software Description

We decided to use a Naive Bayes classifier to accurately predict which age group (18 - 24, 25 - 44, and 45+) will be most likely to read any given Cornell Sun article. A Naïve Bayes classifier is a classifier that uses Bayes' theorem applied to a series of independent assumptions between the features. The formula for Bayes' theorem is given as follows:



The diagram shows the formula for Bayes' theorem:  $P(c | x) = \frac{P(x | c)P(c)}{P(x)}$ . Arrows point from labels to parts of the formula: 'Likelihood' points to  $P(x | c)$ , 'Class Prior Probability' points to  $P(c)$ , 'Posterior Probability' points to  $P(c | x)$ , and 'Predictor Prior Probability' points to  $P(x)$ .

$$P(c | X) = P(x_1 | c) \times P(x_2 | c) \times \dots \times P(x_n | c) \times P(c)$$

By applying this method to a large number of independent features from the article and the information we have on current article readership from the Cornell Daily Sun, we can create a robust model that will be able to predict future instances of article readership by hypothesis testing against our model's training data.

In order to train the model, we used data from Cornell Sun's Google Analytics page, which contains data regarding its most popular articles, views per article, and audience. We used audience and webpage data starting from November 8, 2016 to November 29, 2017, which contained a list of article titles and the number of clicks each article received from each age group; however, since there were significantly fewer readers in some of Google Analytics' preset age groups of 18-24, 25-34, 35-44, 45-54,

and 55-64, we grouped together some of the age groups in order to better distribute the number of readers in each group per article. After calculating the distributions of various age groups, we decided to focus on three: 18-24, 25-44, and 45+. However, by adding session counts for each of these 3 buckets, the classifications were too skewed against the mid-range. Since there were not enough entries in the 25-44 bucket relative to the 18-24 and 45+ buckets, no articles were classified as 25-44. To combat this, we instead count the number of sessions within the smaller buckets (18-24, 25-34, 35-44, ...) and whichever bucket has the most sessions dictates which of the 3 larger buckets the article is classified as. Using this method, we were able to have articles classified in each larger bucket. Once we gathered this information from the Google Analytics, we then used this as our training data along with our feature set as inputs into our Naive Bayes model.

In Naive Bayes classification, the feature set must consist of independent features, which means that they should not be correlated in any way. We decided to use word counts and their frequencies as the features our classifier would observe. Once we received all of the titles and article content from the Cornell Sun's Wordpress API in Python, we compiled the word counts by splitting the article body into its word components and then compiling it into a dictionary mapping each string to the count of how many appearances it makes in the article. Once we compiled all of this information for each article, we then fed seventy percent of it into our Bayesian model and trained it to classify articles from the Sun.

We decided to use a Bayesian model implemented in Swift in an iOS application. Our decision to do this was based on our experience in iOS development as well as our

involvement with creating a mobile application for the Cornell Sun itself. Additionally, we created an intuitive mobile prototype for our presentation that could be translated into our work on the Cornell Sun's iOS mobile application. For our prototype application, we read in the text files generated from the Python script and imported them into our Bayesian model. From there, we fed in all of the training data and its predetermined age classification to train the model. In total, we had approximately 819 articles, 574 of which were used as training data for our classifier. Then, we began to test our training model using the remaining thirty percent (245 articles) of our classification data to determine how accurately it can predict the age readership of articles where the information is already determined. From this, it was determined that our model has a 76% accuracy rate when it comes to classifying the average age readership of an article. When we increased the amount of training data to 85% and tested on the remaining 15%, the accuracy rate increased to 78%. After testing variable amounts of training data and the corresponding accuracies (as shown in the graph below), there is a clear correlation between the amount of training data used and the accuracy of our model. Therefore, if we continue to test with increasing amounts of article training data, our model will increase in accuracy and become better at classifying articles.

The Accuracy of our Naive Bayes Model



The prototype used in demonstration is a native iOS application that was built in Swift. In our prototype, we use our existing Cornell Sun news feed as the main screen. From there, a user can then tap on an article and the app will fetch the title of the article and its corresponding post content from the Cornell Sun's Wordpress API. Then, this content will be input into the Bayesian classifier, which will predict which age category would be most likely to read the selected article. The predicted age group will be displayed as a second screen that pushes on top of the news feed with the corresponding article title. From there, this screen can be dismissed and the user will be taken back to the news feed with the latest news and updates from the Cornell Sun's website. A potential extension of this prototype that we are considering building out in the future will take into account the current user's age and train the model while the user browses articles on the app. In order to take into account the user's age, a pop-up message would appear on the screen for the first time the user opens the app where they can select their age category as specified by our training model. Once they select their age category, we would train the model further by using the user's article clicks as training data. This would then tailor the model to not only their age category, but to their personal preferences in article browsing.

## **AI Evaluation**

According to the American Press Institute, older Americans tend to pay closer attention to local politics and foreign issues, while younger Americans tend to follow entertainment and social issues. Prior to training our model, we anticipated that these

trends will reflect in the Cornell Sun's readership as well. However, because the Cornell Sun is targeted towards the entire Cornell community, these genres may be interpreted differently as local news, events, and politics greatly impact the Cornell community. For example, the assault in Collegetown from earlier this year is considered local news, but was the most popular article over this past year among all readers of all ages.

Furthermore, the general engagement on the Cornell Sun is generally more skewed towards a younger population, as the Sun is completely written and operated by Cornell undergraduates. Therefore, some of their writing styles and interests will reflect in the content they produce. Additionally, it is very possible that there is a correlation between the words used in an article and the age group that reads them, as younger adults are more likely to read more casual articles that use different terminology, or similarly, older adults are more likely to read articles that have more formal intonations.

The classifying model we used was a Naive Bayesian model, which is arguably the simplest classifier to implement and understand; however, it might not necessarily be the most appropriate classifier for this instance. We chose Naive Bayes as a classifier because of the abovementioned reasons, as well as the fact that Naive Bayes has also been used in instances such as spam detection and categorizing documents as pertaining to a certain category, which we felt were similar to what we are aiming to do in this project. However, when it comes to news articles, there may have been several instances where the attributes in the feature set were not necessarily independent, which could hinder the classification ability of Naive Bayes. For example, there is significant overlap between word contents in many of the articles, such as frequent words like "the",

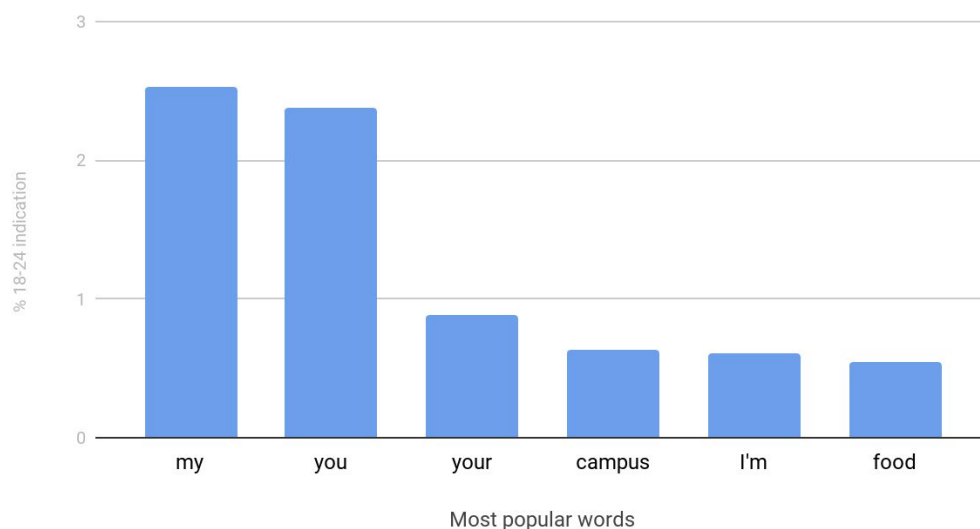
“and”, and “a”, which could be regarded as insignificant data that could accidentally alter Bayes’ theorem. Additionally, the types of words used in certain articles may be correlated to one another, as different categories of articles will be more likely to use certain language: for example, the word “Cornell” and “campus” would be more likely to show up in an article together than two completely unrelated words. This goes against the fundamental principle that supports the Naive Bayesian classifier, as it is heavily assumed that features are independent to one another. On the other hand, spam detection and document classification both use text as the feature set with Naive Bayes and have been very successful, so we do not believe that this is a large cause for concern.

The sample data from the Google Analytics may not be completely representative of each age group, as the readership of the Cornell Sun is largely undergraduates between the ages of 18-24. Therefore, our model is trained mostly on classifying articles as having a likely audience within that age range. One adjustment that could be made to normalize this dataset or improve the model’s predictive capabilities would be to either specifically filter for articles that are read more often by the other age groups such that the distribution between articles and age groups is more evenly spread out or to increase the sample size retrieved from Google Analytics. The advantage of the first possibility is that the model would have an equally distributed training set, which would enable it to have a more egalitarian approach to the new articles it sees and classify articles in each age group more evenly.

## Interpretation of Results

We performed analysis on which words that each age group was more likely to read based on the articles and contents that our model was trained on. From this, we gathered enlightening data on the types of distinct words each age group was more likely to read. We determined this by using our Bayesian classifier to determine the probability of a word showing up in an article given that it was most likely to be read by each of the age categories, and then subtracting the differences between the probabilities to see which age category had a difference of more than 0.05 over the other two age categories. This determined which age category was more likely to see which words, and the results are posted below:

Most Popular Words Read By Ages 18-24



The y-axis in the above graph is computed by taking the probability of seeing the word on the x-axis given the 18-24 classification and subtracting it from the equivalent



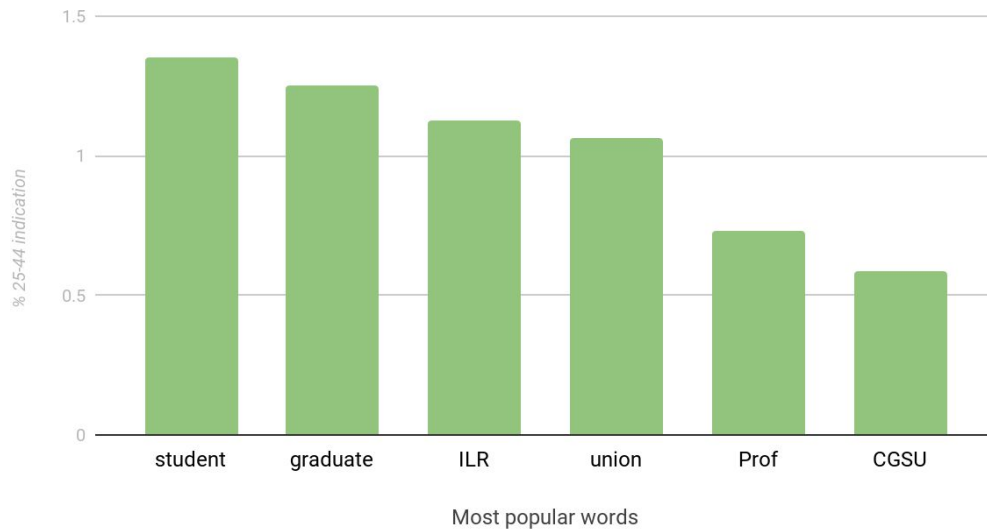
probabilities for the other classifications. To be exact, it can be expressed in the equation below:

$$\%indicator_{18-24} = Pr(word | class_{18-24}) - Pr(word | class_{25-44}) - Pr(word | class_{45+})$$

For clarity, word denotes the word being evaluated on the x-axis, class represents the word being classified by the subscripted age group, and %indicator is the value on the y-axis in the above graph. By comparing this value to 0.05, we determined this value to indicate the 18-24 age class more than both the 25-44 class and the 45+ class combined. Thus, these words are the most likely indicators for the 18-24 classification and dictate trends of articles more likely to be read by the younger audience rather than an older audience.

Looking at the words in the above graph, we noticed that articles written with casual first or second person language were good indicators of an article being read by readers in the 18-24 age range. Since the writers for the Cornell Daily Sun are also undergraduate students, younger readers may identify more with the content of the articles than older readers. Looking at more words that were likely indicators of this age range, terms involving food, Cornell, and social issues on campus were more likely to turn up. This means that younger readers are more likely to click on an article pertaining to issues on campus, food and dining, or social issues pertaining to them.

Most Popular Words Read by Ages 25-44

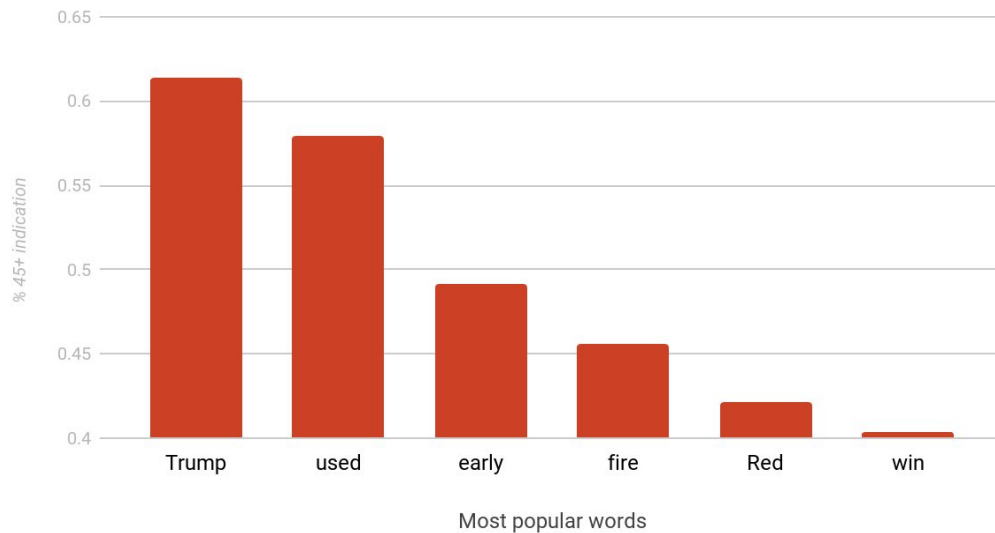


The y-axis in the above graph is computed by taking the probability of seeing the word on the x-axis given the 25-44 classification and subtracting it from the equivalent probabilities for the other classifications, which is expressed in the equation below:

$$\%indicator_{25-44} = Pr(word | class_{25-44}) - Pr(word | class_{18-24}) - Pr(word | class_{45+})$$

Cornell Sun readers in the age range of 25-44 tend to view more articles related to graduate student issues, academics, and unionization according to our analysis. Because the graduate student population does fall into this age bracket, it makes sense that words such as “graduate” and “CGSU” are more frequently read, as graduate students are concerned about campus issues and policies that are relevant to them. Furthermore, with the recent talks of unionization and its relevance to professionals in the ILR school, the prevalence of “ILR” and “union” make sense here as well. The tendency to read fewer of the personal or food articles is clear here, as graduate students have different tastes in articles compared to undergraduates.

### Most Popular Words Read by Ages 45+



The y-axis in the above graph is computed by taking the probability of seeing the word on the x-axis given the 25-44 classification and subtracting it from the equivalent probabilities for the other classifications, which is expressed in the equation below:

$$\%indicator_{45+} = Pr(word | class_{45+}) - Pr(word | class_{18-24}) - Pr(word | class_{25-44})$$

According to the above graph, older readers ages 45 and above tend to read more about politics, Trump, and athletics. As shown by last year's presidential election, Trump voters tend to be in the older age bracket, so they have followed his candidacy more closely on the Sun. Words such as "Red" and "win" stem from Cornell's Big Red sports program, which potentially signals that older alumni are reliving the glory days by closely following Cornell athletics. We can't exactly pinpoint why other words such as "used", "early", and "fire" are more prevalent in articles that are more likely to be read by this age group.

## Conclusion

In conclusion, we successfully used a Naive Bayesian classifier to determine the typical readership of a given article on the Cornell Sun. Our training data was based on the words and their frequencies in a given article, and classified them into three groups: 18-24, 25-44, and 45+. We found that our model has an accuracy of 76%, which can be improved by expanding the dataset. From there, we used our model to test novel articles and categorize them into their prospective age groups and the words that each age group was most likely to read. We discovered that there is a correlation between age groups and the words and article types that they read. We hope that these findings will be useful to the Cornell Sun and impact their article publication strategies in the future.

## References

Swift App Github Link: <https://github.com/cornell-sun/sun-article-age-classifier>

Training Data Github Link: <https://github.com/cornell-sun/sun-classifier-training-data>

<https://www.americanpressinstitute.org/publications/reports/survey-research/social-demo-graphic-differences-news-habits-attitudes/>

<https://blog.bufferapp.com/the-most-popular-words-in-most-viral-headlines>

<http://www.journalism.org/2016/07/07/pathways-to-news/>