


# Dating Profile Age Analysis – a Supervised Learning Application

Rushil Singh

## Related papers

[Download a PDF Pack](#) of the best related papers 



[Arabic text classification using master-slaves technique](#) Arabic text classification using mas...  
Zinah Abdulridha Abutiheen

[Abutiheen\\_2018\\_J.\\_Phys.%3A\\_Conf.\\_Ser.\\_1032\\_012052.pdf](#)  
Zinah Abdulridha Abutiheen

[A Data Augmentation Approach to Short Text Classification](#)  
Ryan R Rosario

# Dating Profile Age Analysis – a Supervised Learning Application

Arielle Friedman, Rushil Singh, Ao Ao Feng

## Abstract

We designed an agent that uses supervised machine learning and word frequency analysis to analyze people's dating profiles. It classifies what words people in different age groups tend to use in their profiles, and can predict the age group of a dating profile's writer. We programmed our agent in Java and used TF-IDF scores and multinomial logistical regression. This technique has important potential market and research applications.

## 1. Introduction

More and more phone apps, dating sites like OkCupid, and social media sites like Facebook rely on user generated data to deliver customized services. This data tends to be gathered through Apps, quizzes and other measures that require active user participation. However, information gleaned through self-surveys have well-documented shortcomings (discuss cogs paper).

Could user information be gathered in ways that didn't bear these shortcomings? One way this could be done is through Natural Language Processing (NLP). NLP is the study of syntax, words and grammars to gather information. It has been used in a wide variety of research applications for a broad range of purposes. We will use NLP to design an App which gathers information on how people in different age groups use language differently to describe themselves on an online dating site. The App will also guess the age range of novel users, which helps us test the App and provides an entertaining way to engage users.

We've decided to use a particular kind of NLP called word frequency analysis, which only takes into account word frequencies. This approach has obvious shortcomings – much of the meaning of language emerges from elements which are not included in word frequency analysis, like sentences, context, phrases, idioms and even compound words – the compound “hot dog” for example will lose all meaning if analyzed in terms of the words “hot” and “dog”. When performed on large quantities of data however, word frequency analysis preserves semantic meaning a statistically significant enough portion of the time to provide useful information. Furthermore, word frequency analysis is easy for an AI system to perform accurately on huge bodies of text, whereas teaching an agent more complex language structures is more difficult and error-prone. We'll look at a few examples where word frequency analysis was used successfully on a large data set.

Vongpumivitch et al. (2009) used word frequency analysis to determine how often applied Linguistics researchers used words from the Academic Word List (AWL) in their publications. Academic language is often one of the biggest challenges facing academic teachers and learners of English as a foreign language, so this study has important pedagogical implications. They found that AWL words accounted for 11.7% of all words used in the entire Applied Linguistics Research Corpus, a data source with over 1.5 million words. This study demonstrates the scale on which word frequency analysis can be used and its potential scope in gathering broad statistical data on huge amounts of textual information.

Baruch Vilensky (1996) used word frequency analysis to determine the relative word choice

distance between authors. He found that an agent using word frequency analysis alone could successfully tell whether or not two books were written by the same author. This shows how word frequency analysis can be used to gather information about writers which may not be superficially obvious.

Coyle et al. (2012) designed an agent that performed word frequency analysis on drug testimonials that people posted on the website Erowid. These documents are likely in a straightforward reading to fall into many of the shortcomings of self-reports, much like the dating profiles we're looking at, but through word frequency analysis we can probe these documents for information that the users didn't mean to convey. Using word frequency analysis, Coyle's agent was able to accurately determine which drug was associated with each testimonial, and could identify words that were strongly associated with certain kinds of drugs.

For our “Guess your age” dating profile App, we've decided to use supervised machine learning, TF-IDF scores, and multinomial logistical regression (MLR) to carry out word frequency analysis. We'll go on to explain why we made those choices, and how we carried them out.

## **2. Methods**

We designed our agent to carry out supervised machine learning. This means that the agent will be trained on a data set where for each data point it is given the dependent variable (the age category) and the independent variable (the dating profile), and then will be tested on a data set where it is given the independent variable (the dating profile) and has to predict the dependent variable (the age category). Unsupervised machine learning is where the agent is never given labelled data and has to infer patterns on its own. We chose supervised machine learning because supervised machine learning can be done on much smaller data sets. We gathered about 700 data points (dating profiles) and we would have needed tens of thousands at least to carry out unsupervised machine learning. An unsupervised machine learning system would be able to find patterns that are not known to the experimenters, so this may be an interesting direction to take in the future.

We combine two techniques for the machine learning component of our project, TF-IDF (i.e. term frequency-inverse document frequency), and multinomial logistic regression (MLR). TF-IDF scores calculate the relative frequency of each word in a document. The TF score is the frequency of each word in its particular document. This alone would not be enough – many words appear frequently simply because they're common English words, like “and”, “I”, and “in”. We're looking for words which not only appear frequently, but which appear frequently in this document relative to other documents. That's why we use IDF scores. The IDF value is the proportion of profiles a word appears in. TF and IDF scores are multiplied together to give a score for each word in each document which accurately reflects the relative frequency of that word in that document – how “significant” it is.

MLR allows us to predict the likelihood of a categorical dependent variable based on independent predictor data (generated from TF-IDF). MLR is a generalized form of logistical regression used when the dependent variable has more than two categories. MLR is what allows the agent to predict the age category of an unknown profile based on the training data it has processed.

One of the most common applications of TF-IDF is in web searches. Rahman et al. (2013) studied

the usefulness of TF-IDF scores in dealing with topic drift in web searches. Topic drift is when the content of a search result drift away from the topic in its heading. They designed a webpage ranking system which uses TF-IDF to improve the relevancy of search results, demonstrating the usefulness of TF-IDF scores in pruning irrelevant data.

Nor is the utility of TF-IDF limited to written text. Smith et al. (1997) used TF-IDF to design a video search program that reduces each video to a 2 minute fragment to speed up search time. These fragments preserve relevant audio keywords while eliminating search-irrelevant information. This is accomplished using TF-IDF scores, demonstrating the power of this tool in a variety of different modalities.

For our data set, we used 700 profiles from friendfinder.com. We copied the text of the self-description section of each person's profile (labelled "Introduction" on the site) and recorded the person's age. We divided the profiles into three age groups and designed our program around those categories. We programmed our agent in Java.

During the training phase, the agent applies TFIDF scores and Multinomial Logistical Regression to a large body of training data to determine the weight of each word relative to each age category. During the testing phase, the agent uses these values to assign the profile being tested a "score" for each category, and selects the category with the highest score. The dating profile's writer most likely falls into this age range category.

We chose the age ranges [20 – 29], [30 – 39], and [40+]. The techniques we used are scalable, so we could increase the number of age groups for future versions. Initially we wanted the agent to guess the participant's exact age, but if we used our current approach to do so, we'd risk over-determination. Over-determination is the risk of information loss when the categories are small and overlapping. If someone uses words that are common to a 28-year-old, they are more likely to be 29 than if they use words that are common to a 58-year-old. But if we use a categorical analysis, that kind of information is lost. In future versions, we hope to program our agent to guess participants' exact age by programming our agent to use ordinary least squares regression, which takes a continuous dependent variable.

The use of categories also offsets the issue of how we can be sure that people are reporting their ages honestly. If we were using exact ages, it would be crucial to know that the ages were accurate, but since we're using categories, the ages can be approximate. It doesn't matter at all if people lie within a category, but if they lie across categories in a systematic way, it could influence our results. This may be taking place, since people may lie to make themselves appear to be in a particular decade. In future versions we could offset this by choosing age categories which don't correspond to decades.

## 2.1: Calculating TF-IDF scores

TF-IDF calculates the relative frequency of each word in each profile. The TF value equals the number of times the target word appears in that profile divided by the total number of words in that profile. The IDF value is the proportion of profiles a word appears in. The agent calculates this by dividing the total number of profiles by the number of profiles that the target word appears, and then take the tenth log of the result.

## 2.2: Performing MLR

To perform MLR, the agent needs to have three sets of Beta-values corresponding to each of the three age range categories. The Beta-values for a category are what we've chosen to call the representation of the linear weight of each word relative to that category – a rough estimate of how likely that word is to indicate that category.

## 2.3: Testing and Assignment

To determine the dating age of a profile, the agent calculates its scores for each of the age categories. The score for a profile, for each category, is the dot product of the TF-IDF values for all the words in that profile, with the Beta-value of each word for that category:

This yields a number between -1 and +1 for each profile for each category. The highest scoring age category is chosen as the most likely candidate for that profile.

## **3. Evaluation**

To test our agent's accuracy, we performed a 5-fold cross analysis. We divided our data of 700 profiles into 5 equal parts of 140 profiles each, all of which contained some profiles from each of the three categories. We cycled through using each 1/5th as our testing examples and training on the remaining 4/5th, or 560 profiles. Our data is displayed below.

Our agent was accurate ~54.4% of the time. Since each profile could be classified into 3 possible categories, we would expect our program to have 33.3% average accuracy by chance, with a standard error of roughly 4% (on 140 test cases). Our program's accuracy was well above the margin of error of what would be expected by chance.

## **4. Results**

Our agent can tell us which words corresponded to each age group most closely. Common words in the [20 – 29] age group included “studying”, “ego”, “chill”, “creepy”, “dude”, and “gay”. For the [30 – 39] age group, “therapy”, “fantasies”, “grateful”, “hollywood”, and “unpredictable”. And for [40+], “financially”, “secure”, “prince”, “nonsmoker”, “worker”, “sexual” and “sincerity”. We graphed some of our more interesting results below. Positive Beta-values indicate that a word is particularly common for that age category, vice versa for negative.

## **5. Conclusion**

This technique we were able to generate interesting and accurate results about language differences across generations. In future, we could improve our program's accuracy and utility by employing feature elimination, larger sample sizes across different dating sites, and a larger number of category gradations (maybe a category for each five year age range instead of each decade). Word frequency analysis has some drawbacks in that it can't analyze text on the level of phrases and sentences, and it's insensitive to the multiple and potentially shifting meanings of individual words. Nonetheless, we have demonstrated how this deceptively simple tool has the potential to generate interesting, subtle and powerful results.

## **6. Acknowledgements**

We would also like to thank our T.A. David Buchman for pointing us in the right direction and helping to clarify our original idea. We would like to thank our professor Dr. David Poole for getting us inspired about machine learning.

## 7. References

- Coyle, J. R., Presti, D. E., & Baggott, M. J. (2012). Quantitative analysis of narrative reports of psychedelic drugs. *Cornell University Library*, doi: <http://arxiv.org/abs/1206.0312>
- Rahman, M., Ahmed, S., Islam, S., & Rahman, M. (2013). An effective ranking method of webpage through tf-idf and hyperlink classified pagerank. *International Journal of Data Mining & Knowledge Management Process*, 3(4), 149-156. doi: 10.5121/ijdkp.2013.3411
- Smith, M., & Kanade, T. (1997). Video skimming and characterization through the combination of image and language understanding technique. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 775 - 781. doi: 10.1109/CVPR.1997.609414
- Vilensky, B. (1996). Can analysis of word frequency distinguish between writings of different authors?. *Physica A*, 231, 705-711.
- Vongpumivitch, V., Huang, J., & Chang, Y. (2009). Frequency analysis of the words in the academic word list (awl) and non-awl content words in applied linguistics research papers. *English for Specific Purposes*, 28, 33-41. doi: 10.1016/j.esp.2008.08.003

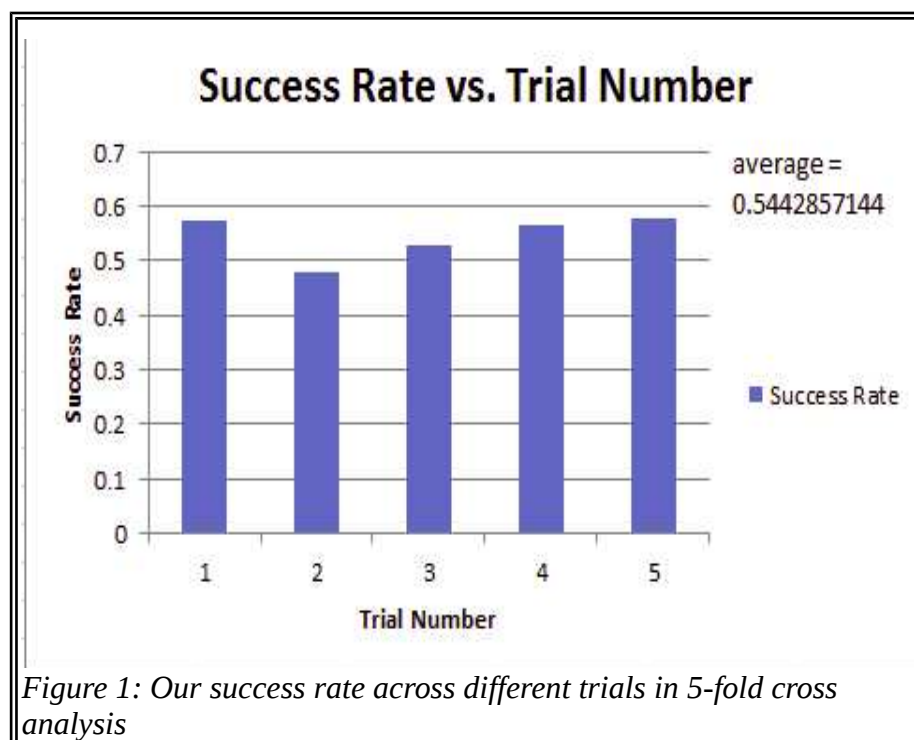


Figure 1: Our success rate across different trials in 5-fold cross analysis

Appendix: Figures

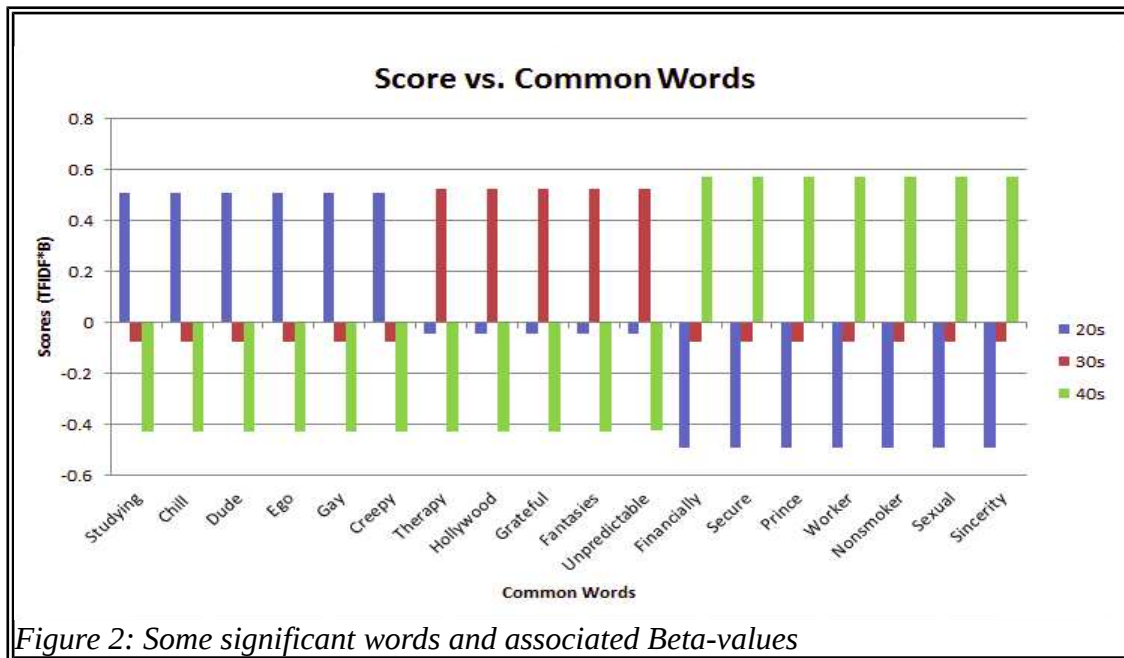


Figure 2: Some significant words and associated Beta-values

