# Predicting Speed-Date Outcomes with Conversational and Network Features

Emma Pierson
emmap1@stanford.edu

Andrew Suciu
asuciu@stanford.edu

*with many thanks to Professor Dan Jurafsky*[*] *of Stanford University for advice and data*

## 1 Introduction

**Data:** In 2005, researchers conducted three evenings of graduate-student speed dating. During each evening, twenty single women and twenty single men each went on a four minute "date" with every participant of the opposite sex. After each date, each participant filled out a survey indicating whether they were interested in the other person (if the other person was also interested, the students eventually received each others email–this was the motivation for participating in the study) and made a subjective rating from one to ten of the following **response features**: how *friendly, flirtatious, awkward*, and *assertive* they were; how *friendly, flirtatious, awkward, assertive, attractive, sincere, intelligent, funny, ambitious, and courteous* the other person was; how much they *enjoyed the date*; how much information they shared; and how much they felt that the two of them *clicked*. For each date, from each participant's perspective, we have the **outcome** variable of whether they were interested, the response variables, and both audio and written transcripts. Additionally, Dan Jurafsky and colleagues extracted a set of **conversational features** from the date transcripts which are too numerous to list, but which include *instances of laughter, interruption, questions*, etc.

**Motivation:** Ranganath, Jurafsky, and McFarland have written two papers ([5], [6]) using this data set so far, both of which used support vector machines trained on the conversational features to predict the response features. He told us that they were having trouble predicting the outcome variable, especially for men, above chance. Predicting whether or not a given person is romantically interested in another is a difficult but rewarding task. We are typically reserved when it comes to displaying open romantic interest toward strangers for social reasons, so being able to predict interest from conversational data could reduce the number of lonely grad students–a noble end in itself. More broadly, while these dating situations are contrived, we think our project will offer insight into other dating situations (like online dating websites), social networks, and perhaps even ordinary conversation.

Our novel approach to the data is to predict the outcome of a date using not just the conversation between the two people, but also other dates involving the subject or target and similarities between the subject and the group. In short, while Jurafsky's interest was primarily NLP, we are studying the network structure of the problem, attempting to predict edges in a directed, bipartite graph. This research is also potentially applicable to other social network problems–predicting friendships

[*]jurafsky@stanford.edu

on Facebook, followers of political candidates, businesses, or blogs, citations in academic research, etc. We lack the background for feature analysis, but examining the importance of various features in the model could also be useful.

## 2 Conversational Features

We separated men and women into separate groups and used linear regression and support vector machines (SVMs) with linear, rbf, and polynomial kernels to predict the outcome of each date based only on conversational features, measuring success with leave-one-out-cross-validation (LOOCV). When we trained on the full dataset, there was low variance, but our training results were around 60% for both genders. Simply predicting "success" for every man's date and "failure" for every woman's date yields accuracy rates of 56.3% and 62.6%, respectively, so we needed a more sophisticated approach.

At the other extreme, training one model for each individual on only the dates they experienced produced the opposite problem. With 20 data points for each person and 37 features, all models overfit the data. The averaged LOOCV errors over all of the men and over all of the women exhibited high variance, fitting the training set almost perfectly, but failing to generalize.

To compromise between these extremes, we needed to preserve some notion of individual differences in dating preferences while making use of the information resulting from the collective dating experiences. We trained individual models for each person on the entire dataset for their gender, but weighted the dates the individual went on more heavily. Weighting the individual's dates, ($M_i$, $F_i$)[1], 5-10 times as heavily as the other dates, ($M_j$, $F_j$), produced error rates far better than chance.

This weighting scheme does not acknowledge that some dates between ($M_j$, $F_j$) may be more relevant to a ($M_i$, $F_i$) based on a similarity between the $M_j$ and $M_i$. We computed similarity matrices between $M_j$ and $M_i$ in two ways: we looked at similarity in terms of behavior on the dates, and in terms of liking the same people, created rankings for each other participant of the same gender, and weighted their dates proportionally.

Our most successful behavior-similarity metric came from using the cosine of the vocabulary vectors, each row indicating the number of times a given word was said, between an $M_i$ and each $M_j$, weighted by inverse document frequency; effectively, TF-IDF.

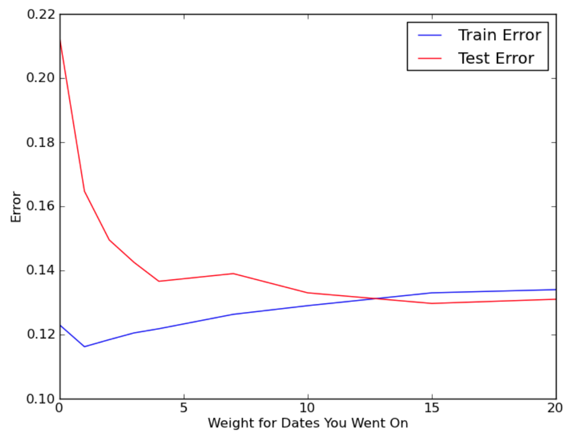
We compared different preference rankings (including simple indicators, Jaccard index, common neighbors) and had best results perturbing the weights according to $\sum_{F_j}(1(\textit{i and j agree about an } F_j)/1(\textit{i and j dated the same } F_j)$.

## 3 Outcome Features

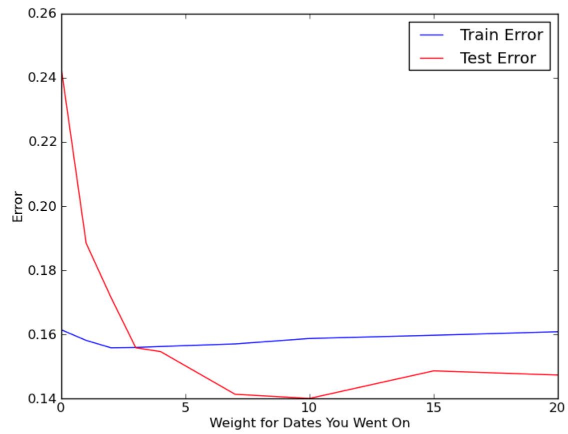
Outcome features intuitively contain more information about a date than conversational features. As a subject is rating their target after the date, they have solidifed their opinion of the other person to the point where their ratings will likely be colored by their choice of outcome. Using an approach similar to the one outlined above with normalized outcome features, we were able to predict date outcomes with high accuracy, despite the subjective noise in these features. The figures

[1] A note on notation: throughout this paper, we use ($M_i$, $F_i$) to denote the test date and ($M_j$, $F_j$) to denote other dates not involving the same $M_i$. We also refer to things from the male perspective, which is not an attempt to reinforce patriarchal hegemony but merely an attempt at consistent notation.

below demonstrate the result of training on the full dataset while increasing the weight for dates that subject $i$ went on.




**Women**




**Men**

We attempted to reduce the noise in the ratings by representing each score as a normalized value for each person (i.e. individualized standard deviations from a mean of zero), but this did not increase the strength of the model.

# 4 Network Features

After we found, from the weighted samples, that the date between two other subjects ($M_j$, $F_j$) was useful for predicting the outcome of a date between a subject and a target of interest ($M_i$, $F_i$), it seemed logical to predict the outcome of ($M_i$, $F_i$) without using any features of that particular date at all. Could we predict whether $M_i$ would like $F_i$ before they even met, based only on the network of who had liked whom in the other dates? This is a specific instance of link prediction for which numerous algorithms have been developed [1][2][3][4][7].

Because of the special structure of our network (bipartie, directed, and very small), many of these methods are not applicable; in particular, widely used features like common neighbors must be adapted for bipartite graphs (since *common neighbors*($M_i$, $F_i$)=0 for all $M_i$, $F_i$). However, because bipartite graphs turn up so frequently (modeling everything from Yelp ratings to DNA-protein interactions), they have nurtured their own subclass of link-prediction algorithms, ranging from the straightforward to the exotic [1][2][7]. We decided to develop an algorithm that was (a) not convoluted (because our dataset was so small) and (b) potentially combinable with our other prediction methods for conversational and outcome features. This ruled out some of the more exotic methods, like learning psuedokernels [1] or learning an embedding function for network nodes [2]. We defined a set of features for each ($M_i$, $F_i$) based on the network topology and used LOOCV with logistic regression and SVMs to predict the presence of edges.

We began by defining the features on an ad-hoc basis–for example, *other guys who liked date* $i=\sum_{M_j}(1(\text{edge}(M_j, F_i)))$, but then, inspired by Leskovec's systematic binary enumeration of feature possibilities in [3], decided to do things more systematically. Extending [1], for each edge ($M_i$, $F_i$), we removed the edge and then defined four projected graphs that were undirected but weighted. Notating a graph by $\langle$*vertices*, *edge weights*$\rangle$, the first projected graph was given by $\langle$Males, *common neighbors*($M_i$, $M_j$)$\rangle$–intuitively, a graph where the weight of an edge between two males was how many females they liked in common.

We defined a second graph, *M: who likes you*, by replacing how many females they liked in common with how many females they *had been liked by* in common. We defined the projections for the females similarly. To normalize edge weights, we defined a second set of four projected graphs where, rather than using common neighbors, we used the Jaccard coefficient (our original weight divided by the size of the union of the relevant edge sets). This projections generalized those in 1, which were binary and not applicable to directed graphs. We defined two features from each projection. For the male graphs, for example, our features were $\sum_{M_j}(w_{ij}*\text{edge}(\text{M}_j,\ \text{F}_i))$ and $\max_{M_j}(w_{ij}*\text{edge}(\text{M}_j,\ \text{F}_i))$, where $w_{ij}$ was the weight of the edge between $\text{M}_i$ and $\text{M}_j$ in the projected graph, and edge was +1 if the edge was present in the original graph and -1 if it was not. Similarly, for the female graphs, our features were $\sum_{F_j}(w_{ij}*\text{edge}(\text{M}_i,\ \text{F}_j))$ and $\max_{F_j}(w_{ij}*\text{edge}(\text{M}_i,\ \text{F}_j))$. We did not expect all 16 features to work well, but the systematic enumeration allowed us to see which binary choices worked better. We defined two additional features, *she likes you*=edge($\text{F}_i$, $\text{M}_i$) and *num you like*=$\sum_{F_j}(\text{edge}(\text{M}_i,\ \text{F}_j))$, where edge in this case was boolean, not +/-1. By training on subsets of these 18 features and comparing the error rates, we derived a number of results:

1. Whether you like someone is correlated with whether they like you.
2. Using sum works considerably better than using max–intuitive, since max comes from only one edge and should be noisy.
3. The *who you like* features were marginally more predictive than the *who likes you* features. We thought that femalewholikesyou and malewhoyoulike should be most predictive for male-female edges, but this turned out not to be the case.
4. Jaccard and common neighbors were roughly equally predictive–normalization turned out not to matter.
5. Male projection features were marginally more predictive for male-female edges than were female projection features, and vice versa. Your companions evidently offer more insight than your dates.

Using all 18 features overfits the dataset, and since the only major performance difference between our binary choices was that sum significantly outperformed max, we threw out the max features for a final feature set size of 10. (An extension of this project would involve more sophisticated feature selection, whether by forward search or some adaptation of mutual information.) This produced an test accuracy of 82.2% for men and 82.5% for women with LOOCV SVM with RBF kernel.

# 5 Multiple Feature Domains

We attempted to combine features from both the network and conversational domains to improve predictions without resorting to looking at the outcome features–which we consider tantamount to predicting how the date went after asking the person how it went. Without sophisticated feature selection techniques, we achieved best results by combining the five conversation features with the largest absolute valued weights *"hate", "sex", "you know", "question", "approval"*, with the network features. However, further research should consider more rigorous methods of combination.

# 6 Results

| **Mean LOOCV Prediction Accuracy** (unless noted, best results were obtained with a SVM using RBF kernel) | | |
|---|---|---|
| | **Men** | **Women** |
| Baseline | 56.3% | 62.6% |
| Conversation Features (full dataset, unweighted) | 61.0% (logistic regression) | 65.0% (logistic regression) |
| Conversation Features (preferential weights for you, uniform weights for everyone else) | 63.8% | 68.7% |
| Conversation Features (preferential weights for you, weights proportional to TF-IDF similarity ranking for everyone else) | 65.2% | 69.9% |
| Conversation Features (preferential weights for you, weights proportional to agreement-fraction similarity ranking for everyone else) | 66.0% | 69.0% |
| Outcome Features (full dataset, unweighted) | 82.2% | 83.6% |
| Outcome Features (preferential weights for you, uniform for everyone else) | 85.3% | 86.4% |
| Outcome Features (preferential weights for you, TF-IDF weighted for others) | 84.0% | 87.2% |
| Outcome Features (preferential weights for you, agreement-fraction weighted for others) | 86.0% | 87.3% |
| Network Features | 82.3% | 82.5% |
| Mixture of Conversation and Network Features | 81.3% | 83.5% |

# 7 Conclusion

With adjusted weightings, we were able to successfully predict date oucomes at rates significantly greater than the baseline. We were also able to achieve extremely high performance with network features, even higher than that attained using outcome features, without any information about the date being predicted. However, this comparative success may only reflect our failure to construct an effective model for learning on outcome features.

As mentioned above, we believe this project could be greatly improved with more intelligent feature selection (forward search, an adaptation of mutual information, or some other method). Future experiments might also use deep learning to find better featues or similarity metrics from the topological data.

# References


[1] N. Benchettara, R. Kanawati, C. Rouveirol. *Supervised Machine Learning applied to Link Prediction in Bipartite Social Networks*. Conference on Advances in Social Networks Analysis and Mining, 2010.

[2] J. Kunegis, E. De Luca, S. Albayrak. *The Link Prediction Problem in Bipartite Networks*. IPMU, LNAI 6178, 2010.

[3] J. Leskovec, D. Huttenlocher, J. Kleinberg. *Predicting Positive and Negative Links in Online Social Networks*. International World Wide Web Conference Comittee, April, 2010.

[4] D. Liben-Nowell, J. Kleinberg. *The Link Prediction Problem for Social Networks*. Proceedings of the Twelfth Annual ACM International Conference on Information and Knowledge Mangament, 2003.

[5] R. Ranganath, D. Jurafsky, D. McFarland. *It's Not You, it's Me: Detecting Flirting and its Misperception in Speed-Dates*. Unpublished, 2009.

[6] R. Ranganath, D. Jurafsky, D. McFarland. *Detecting Friendly, Flirtatious, Awkward and Assertive Speech in Speed-Dates*. Unpublished, 2010.

[7] Y. Yamanishi. *Supervised Bipartite Graph Inference*. NIPS, 2008.