It's Not You, it's Me: Detecting Flirting and its Misperception in Speed-Dates

Rajesh Ranganath
Computer Science Department
Stanford University
rajeshr@cs.stanford.edu

Dan Jurafsky
Linguistics Department
Stanford University
jurafsky@stanford.edu

Dan McFarland
School of Education
Stanford University
dmcfarla@stanford.edu

Abstract

Automatically detecting human social intentions from spoken conversation is an important task for dialogue understanding. Since the social intentions of the speaker may differ from what is perceived by the hearer, systems that analyze human conversations need to be able to extract both the perceived and the intended social meaning. We investigate this difference between intention and perception by using a spoken corpus of speed-dates in which both the speaker and the listener rated the speaker on flirtatiousness. Our flirtation-detection system uses prosodic, dialogue, and lexical features to detect a speaker's intent to flirt with up to 71.5% accuracy, significantly outperforming the baseline, but also outperforming the human interlocuters. Our system addresses lexical feature sparsity given the small amount of training data by using an autoencoder network to map sparse lexical feature vectors into 30 compressed features. Our analysis shows that humans are very poor perceivers of intended flirtatiousness, instead often projecting their own intended behavior onto their interlocutors.

1 Introduction

Detecting human social meaning is a difficult task for automatic conversational understanding systems. One cause of this difficulty is the pervasive difference between intended social signals and the uptake by the perceiver. The cues that a speaker may use to attempt to signal a particular social meaning may not be the cues that the hearer focuses on, leading to misperception.

In order to understand the impact of this difference between perception and intention, in this paper we describe machine learning models that can detect both the social meaning intended by the speaker and the social meaning perceived by the hearer. Automated systems that detect and model these differences can lead both to richer socially aware systems for conversational understanding and more sophisticated analyses of conversational interactions like meetings and interviews.

This task thus extends the wide literature on social meaning and its detection, including the detection of emotions such as annoyance, anger, sadness, or boredom (Ang et al., 2002; Lee and Narayanan, 2002; Liscombe et al., 2003), speaker characteristics such as charisma (Rosenberg and Hirschberg, 2005), personality features like extroversion or agreeability (Mairesse et al., 2007; Mairesse and Walker, 2008), speaker depression or stress (Rude et al., 2004; Pennebaker and Lay, 2002; Cohn et al., 2004), and dating willingness or liking (Madan et al., 2005; Pentland, 2005).

We chose to work on the domain of *flirtation* in speed-dating. Our earlier work on this corpus showed that it is possible to detect whether speakers are perceived as flirtatious, awkward, or friendly with reasonable accuracy (Jurafsky et al., 2009). In this paper we extend that work to detect whether speakers themselves intended to flirt, explore the differences in these variables, and explore the ability and inability of humans to correctly perceive the flirtation cues.

While many of the features that we use to build these detectors are drawn from the previous literature, we also explore new features. Conventional methods for lexical feature extraction, for example, generally consist of hand coded classes of words related to concepts like *sex* or *eating* (Pennebaker et al., 2007). The classes tend to perform well in their specific domains, but may not be robust across domains, suggesting the need for unsupervised domain-specific lexical feature extraction. The naive answer to extracting domain-

specific lexical features would just be to throw counts for every word into a huge feature vector, but the curse of dimensionality rules this method out in small training set situations. We propose a new solution to this problem, using an unsupervised deep autoencoder to automatically compress and extract complex high level lexical features.

2 Dataset

Our experiments make use of the SpeedDate Corpus collected by the third author, and described in Jurafsky et al. (2009). The corpus is based on three speed-dating sessions run at an American university in 2005, inspired by prior speed-dating research (Madan et al., 2005). The graduate student participants volunteered to be in the study and were promised emails of persons with whom they reported mutual liking. All participants wore audio recorders on a shoulder sash, thus resulting in two audio recordings of the approximately 1100 4-minute dates. Each date was conducted in an open setting where there was substantial background noise. This noisy audio was thus hand-transcribed and turn start and end were hand-aligned with the audio. In addition to the audio, the corpus includes various attitude and demographic questions answered by the participants.

Each speaker was also asked to report how often their date's speech reflected different conversational styles (awkward, flirtatious, funny, assertive) on a scale of 1-10 (1=never, 10=constantly): "How often did the other person behave in the following ways on this 'date'?". In addition they were also asked to rate their own intentions: "How often did you behave in the following ways on this 'date'?" on a scale of 1-10.

In this study, we focus on the *flirtation* ratings, examining how often each participant said they were flirting, as well as how often each participant was judged by the interlocutor as flirting.

Of the original 1100 dates only 991 total dates are in the SpeedDate corpus due to various losses during recording or processing. The current study focuses on 946 of these, for which we have complete audio, transcript, and survey information.

3 Experiment

To understand how the perception of flirting differs from the intention of flirting, we trained binary classifiers to predict both perception and intention. In each date, the speaker and the interlocutor both labeled the speaker's behavioral traits on a Likert scale from 1-10. To generate binary responses we took the top ten percent of Likert ratings in each task and labeled those as positive examples. We similarly took the bottom ten percent of Likert ratings and labeled those as negative examples. We ran our binary classification experiments to predict this output variable. Our experiments were split by gender. For the female experiment the speaker was female and the interlocutor was male, while for the male experiment the speaker was male and the interlocutor was female.

For each speaker side of each 4-minute conversation, we extracted features from wavefiles and transcripts, as described in the next section. We then trained four separate binary classifiers (for each gender for both perception and intention).

4 Feature Descriptions

We used the features reported by Jurafsky et al. (2009), which are briefly summarized here. The features for a conversation side thus indicate whether a speaker who talks a lot, laughs, is more disfluent, has higher F0, etc., is more or less likely to consider themselves flirtatious, or be considered flirtatious by the interlocutor. We also computed the same features for the *alter* interlocutor. *Alter* features thus indicate the conversational behavior of the speaker talking with an interlocutor they considered to be flirtatious or not.

4.1 Prosodic Features

F0 and RMS amplitude features were extracted using Praat scripts (Boersma and Weenink, 2005). Since the start and end of each turn were time-marked by hand, each feature was easily extracted over a turn, and then averages and standard deviations were taken over the turns in an entire conversation side. Thus the feature F0 MIN for a conversation side was computed by taking the F0 min of each turn in that side (not counting zero values of F0), and then averaging these values over all turns in the side. F0 MIN SD is the standard deviation across turns of this same measure.

4.2 Dialogue and Disfluency Features

A number of discourse features were extracted, following Jurafsky et al. (2009) and the dialogue literature. The dialog acts shown in Table 2 were detected by hand-built regular expressions, based on analyses of the dialogue acts in the

F0 MIN	minimum (non-zero) F0 per turn, averaged over turns
F0 MIN SD	standard deviation from F0 min
F0 MAX	maximum F0 per turn, averaged over turns
F0 MAX SD	standard deviation from F0 max
F0 MEAN	mean F0 per turn, averaged over turns
F0 MEAN SD	standard deviation (across turns) from F0 mean
F0 SD	standard deviation (within a turn) from F0 mean, averaged over turns
F0 SD SD	standard deviation from the f0 sd
PITCH RANGE	f0 max - f0 min per turn, averaged over turns
PITCH RANGE SD	standard deviation from mean pitch range
RMS MIN	minimum amplitude per turn, averaged over turns
RMS MIN SD	standard deviation from RMS min
RMS MAX	maximum amplitude per turn, averaged over turns
RMS MAX SD	standard deviation from RMS max
RMS MEAN	mean amplitude per turn, averaged over turns
RMS MEAN SD	standard deviation from RMS mean
TURN DUR	duration of turn in seconds, averaged over turns
TIME	total time for a speaker for a conversation side, in seconds
RATE OF SPEECH	number of words in turn divided by duration of turn in seconds, averaged over turns

Table 1: Prosodic features from Jurafsky et al. (2009) for each conversation side, extracted using Praat from the hand-segmented turns of each side.

hand-labeled Switchboard corpus of dialog acts. *Collaborative completions*, turns where a speaker completes the utterance begun by the alter, were detected by finding sentences for which the first word of the speaker was extremely predictable from the last two words of the previous speaker, based on a trigram grammar trained on the Treebank 3 Switchboard transcripts. Laughter, disfluencies, and overlap were all marked in the transcripts by the transcribers.

4.3 Lexical Features

We drew our lexical features from the LIWC lexicons of Pennebaker et al. (2007), the standard for social psychological analysis of lexical features. We chose ten LIWC categories that have proven useful in detecting personality-related features (Mairesse et al., 2007): *Anger, Assent, Ingest, Insight, Negemotion, Sexual, Swear, I, We,* and *You*. We also added two new lexical features: "past tense auxiliary", a heuristic for automatically detecting narrative or story-telling behavior, and *Metadate*, for discussion about the speed-date itself. The features are summarized in Table 3.

4.4 Inducing New Lexical Features

In Jurafsky et al. (2009) we found the LIWC lexical features less useful in detecting social meaning than the dialogue and prosodic features, perhaps because lexical cues to flirtation lie in different classes of words than previously investigated. We therefore investigated the induction of lexical features from the speed-date corpus, using a probabilisitic graphical model.

We began with a pilot investigation to see whether lexical cues were likely to be useful; with a small corpus, it is possible that lexical features are simply too sparse to play a role given the limited data. The pilot was based on using Naive Bayes with word existence features (binomial Naive Bayes). Naive Bayes assumes all features are conditionally independent given the class, and is known to perform well with small amounts of data (Rish, 2001). Our Naive Bayes pilot system performed above chance, suggesting that lexical cues are indeed informative.

A simple approach to including lexical features in our more general classification system would be to include the word counts in a high dimensional feature vector with our other features. This method, unfortunately, would suffer from the well-known high dimensionality/small training set problem. We propose a method for building a much smaller number of features that would nonetheless capture lexical information. Our approach is based on using autoencoders to construct high level lower dimension features from the words in a nonlinear manner.

A deep autoencoder is a hierarchichal graphical model with multiple layers. Each layer consists of a number of units. The input layer has the same number of units as the output layer, where the output layer is the model's reconstruction of the input layer. The number of units in the intermediate layers tends to get progressively smaller to produce a compact representation.

We defined our autoencoder with visible units modeling the probabilities of the 1000 most common words in the conversation for the speaker and the probabilities of the 1000 most common words for the interlocutor (after first removing a stop list of the most common words). We train a deep autoencoder with stochastic nonlinear feature detectors and linear feature detectors in the final layer. As shown in Figure 1, we used a 2000-1000-500-250-30 autoencoder. Autoen-

BACKCHANNELS	number of backchannel utterances in side (*Uh-huh.*, *Yeah.*, *Right.*, *Oh, okay.*)
APPRECIATIONS	number of appreciations in side (*Wow*, *That's true*, *Oh, great*)
QUESTIONS	number of questions in side
NTRI	repair question (Next Turn Repair Indicator) (*Wait*, *Excuse me*)
COMPLETION	(an approximation to) utterances that were 'collaborative completions'
LAUGH	number of instances of laughter in side
TURNS	total number of turns in side
DISPREFERRED	(approximation to) dispreferred responses, beginning with discourse marker *well*
UH/UM	total number of filled pauses (*uh* or *um*) in conversation side
RESTART	total number of disfluent restarts in conversation side
OVERLAP	number of turns in side which the two speakers overlapped

Table 2: Dialog act and disfluency features from Jurafsky et al. (2009).

TOTAL WORDS	total number of words
PAST TENSE	uses of past tense auxiliaries *was, were, had*
METADATE	*horn, date, bell, survey, speed, form, questionnaire, rushed, study, research*
YOU	*you, you'd, you'll, your, you're, yours, you've* (not counting *you know*)
WE	*lets, let's, our, ours, ourselves, us, we, we'd, we'll, we're, we've*
I	*I'd, I'll, I'm, I've, me, mine, my, myself* (not counting *I mean*)
ASSENT	*yeah, okay, cool, yes, awesome, absolutely, agree*
SWEAR	*hell, sucks, damn, crap, shit, screw, heck, fuck**
INSIGHT	*think*/thought, feel*/felt, find/found, understand*, figure*, idea*, imagine, wonder*
ANGER	*hate/hated, hell, ridiculous*, stupid, kill*, screwed, blame, sucks, mad, bother, shit*
NEGEMOTION	*bad, weird, hate, crazy, problem*, difficult, tough, awkward, boring, wrong, sad, worry,*
SEXUAL	*love*, passion*, virgin, sex, screw*
INGEST	*food, eat*, water, bar/bars, drink*, cook*, dinner, coffee, wine, beer, restaurant, lunch, dish*

Table 3: Lexical features from Jurafsky et al. (2009). Each feature value is a total count of the words in that class for each conversation side; asterisks indicate including suffixed forms (e.g., *love, loves, loving*). All except the first three are from LIWC (Pennebaker et al., 2007) (modified slightly, e.g., by removing *you know* and *I mean*). The last five classes include more words in addition to those shown.

coders tend to perform poorly if they are initialized incorrectly, so we use the Restricted Boltzmann Machine (RBM) pretraining procedure described in Hinton and Salakhutdinov (2006) to initialize the encoder. Each individual RBM is trained using contrastive divergence as an update rule which has been shown to produce reasonable results quickly (Hinton et al., 2006). Finally, we use backpropagation to fine tune the weights of our encoder by minimizing the cross entropy error. To extract features from each conversation, we sample the code layer (30 unit layer in our encoder) with the visible units corresponding to the most common word probabilities from that document, creating 30 new features that we can use for classification. The conditional distributions of the first layer features can be given by the softmax of the activations for each gender:

$$p(v_i|h) = \frac{\exp(bias_i + \sum_j h_j * w_{ij})}{\sum_{k \in K} \exp(bias_k + \sum_j v_j * w_{kj})} \quad (1)$$

$$p(h_j|v) = \frac{1}{1 + \exp(bias_(j) + \sum_i v_i * w_{ij})} \quad (2)$$

where K is the set of all the units representing the same speaker as i [1], v_i is the ith visible unit, h_j is the jth hidden unit, w_{ij} is the weight between visible unit i and hidden unit j, and bias$_m$ is the offset of unit m. Intuitively, this means that the probability that a hidden unit is activated by the visible layer is sigmoid of the weighted sum of all the visible units plus the unit's bias term. Similarly, the visible units are activated through a weighted sum of the hidden units, but they undergo an additional normalization (softmax) over all the other visible units from the speaker to effectively model the multinomial distribution from each speaker. Since in a RBM hidden units are conditionally independent given the visible units, and visible units are

[1] The visible unit i models word probabilities of either the speaker or the interlocutor, so the softmax is done over the distribution of words for the speaker that unit i is modeling.

conditionally independent given hidden layer, the above equations completely specify the first layer of the model.

To account for the fact that each visible unit in the first layer contained 1000 observations from the underlying distribution we upweighted our features by that factor. During pretraining the "training data" for the higher layers is the activation probabilities of the hidden units of layer directly below when driven by that layer's input data. The intermediate layers in the model are symmetric where the activation probabilities for both the visible and hidden units are of the same form as $p(h_j|v)$ in layer 1. To produce real valued features in the code layer we used linear hidden units. In addition to the likelihood portion of the objective we penalized large weights by using l2 regularization and penalize all weights by applying a small constant weight cost that gets applied at every update. After training to find a good initial point for the autoencoder we unroll the weights and use backpropogation to fine tune our autoencoder.

While interpreting high level nonlinear features can be challenging, we did a pilot analysis of one of the 30 features fixing a large (positive or negative) weight on the feature unit (code layer) and sampling the output units.

The top weighted words for a positive weight are: *O_did, O_live, S_did, S_friends, S_went, O_live, S_lot, S_wait, O_two*, and *O_wasn't* (S for speaker and O for interlocutor). The top weighted words for a negative weight are: *S_long, O_school, S_school, S_phd, O_years, S_years, O_stanford, S_lot, O_research, O_interesting* and *O_education*. At least for this one feature, a large positive value seemed to indicate the prevalence of questions (*wait, did*) or storytelling (*live, wasn't*). A large negative weight indicates the conversation focused on the mundane details of grad student life.

5 Classification

Before performing the classification task, we preprocessed the data in two ways. First, we standardized all the variables to have zero mean and unit variance. We did this to avoid imposing a prior on any of the features based on their numerical values. Consider a feature A with mean 100 and a feature B with mean .1 where A and B are correlated with the output. Since the SVM problem minimizes the norm of the weight vector, there is a bias to put weight on feature A because intuitively the weight on feature B would need to be 1000 times larger to carry the same effect. This argument holds similarly for the reduction to unit variance. Second, we removed features correlated greater than .7. One goal of removing correlated features was to remove as much colinearity as possible from the regression so that the regression weights could be ranked for their importance in the classification. In addition, we hoped to improve classification because a large number of features require more training examples (Ng, 2004). For example for perception of female flirt we removed the number of turns by the alter (*O_turns*) and the number of sentence from the ego (*S_sentences*) because they were highly correlated with *S_turns*.

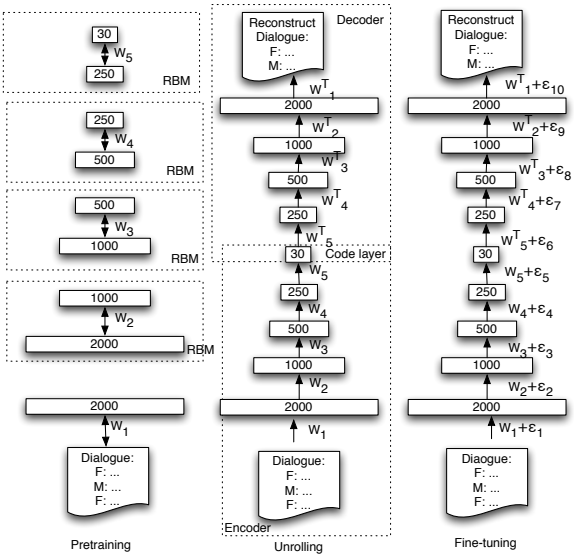

Figure 1: Pretraining is a fully unsupervised procedure that trains an RBM at each layer. Once the pretraining of one layer is complete, the top layer units are used as input to the next layer. We then fine-tune our weights using backprop. The 30 features are extracted from the code layer.

To ensure comparisons (see Section 7) between the interlocutors' ratings and our classifier (and because of our small dataset) we use k-fold cross validation to learn our model and evaluate our model. We train our binary model with the top ten percent of ratings labeled as positive class examples and bottom ten percent of ratings as the negative class examples. We used five-fold cross validation in which the data is split into five equal folds of size 40. We used four of the folds for training and one for test. K-fold cross validation does this in a round robin manner so every example ends up in the test set. This yields a datasplit

of 160 training examples and 40 test examples. To ensure that we were not learning something specific to our data split, we randomized our data ordering.

For classification we used a support vector machine (SVM). SVMs generally do not produce explicit feature weights for analysis because they are a kernelized classifier. We solved the linear C-SVM problem. Normally the problem is solved in the dual form, but to facilitate feature analysis we expand back to the primal form to retrieve w, the weight vector. Our goal in the C-SVM is to solve, in primal form,

$$\min_{\gamma,w,b} \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\xi_i$$
$$\text{s.t. } y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i,\ i = 1,\ldots,m$$
$$\xi_i \geq 0,\ i = 1,\ldots,m \quad (3)$$

where m is the number of training examples, $x^{(i)}$ is the ith training examples, and $y^{(i)}$ is the ith class (1 for the positive class, -1 for the negative class). The ξ_i are the slack variables that allow this algorithm to work for non linearly separable datasets.

A test example is classified by looking at the sign of $y(x) = w^T x^{(test)} + b$. To explore models that captured interactions, but do not allow for direct feature analysis we solved the C-SVM problem using a radial basis function (RBF) as a kernel (Scholkopf et al., 1997). Our RBF kernel is based on a Gaussian with unit variance.

$$K(x^{(i)}, x^{(j)}) = \exp(\frac{-||x^{(i)} - x^{(j)}||^2}{2\sigma}) \quad (4)$$

In this case predictions can be made by looking at $y(x^{(test)}) = \sum_{i=1}^{m} \alpha^{(i)} y^{(i)} \text{rbf}(x^{(i)}, t^{(test)}) + b$, where each $\alpha^{(i)}$, for $i = 1,\ldots,m$ is a member of the set of dual variables that comes from transforming the primal form into the dual form. The SVM kernel trick allows us to explore higher dimensions while limiting the curse of dimensionality that plagues small datasets like ours.

We evaluated both our linear C-SVM and our radial basis function C-SVM using parameters learned on the training sets by computing the accuracy on the test set. Accuracy is the number of correct examples / total number of test examples. We found that the RBM classifier that handled interaction terms outperformed linear methods like logistic regression.

For feature weight extraction we aggregated the feature weights calculated from each of the test folds by taking the mean between them.[2]

6 Results

We report in Table 4 the results for detecting flirt intention (whether a speaker said they were flirting) as well as flirt perception (whether the listener said the speaker was flirting).

	Flirt Intention		Flirt Perception	
	by M	by F	of M	of F
RBM SVM	61.5%	70.0%	77.0%	59.5%
+autoencoder features	69.0%	71.5%	79.5%	68.0%

Table 4: Accuracy of binary classification of each conversation side, where chance is 50%. The first row uses all the Jurafsky et al. (2009) features for both the speaker and interlocutor. The second row adds the new autoencoder features.

In our earlier study of flirt perception, we achieved 71% accuracy for men and 60% for women (Jurafsky et al., 2009). Our current numbers for flirt perception are much better for both men (79.5%), and women (68.0%). The improvement is due both to the new autoencoder features and the RBF kernel that considers feature interactions (feature interactions were not included in the logistic regression classifiers of Jurafsky et al. (2009)).

Our number for flirt intention are 69.0% for men and 71.5% for women. Note that our accuracies are better for detecting women's intentions as well as women's perceptions (of men) than men's intentions and perceptions.

7 Feature Analysis

We first considered the features that helped classification of flirt intention. Table 5 shows feature weights for the features (features were normed so weights are comparable), and is summarized in the following paragraphs:

• Men who say they are flirting ask more questions, and use more *you* and *we*. They laugh more, and use more sexual, anger, and negative emotional words. Prosodically they speak faster, with higher pitch, but quieter (lower intensity min).

[2]We could not use the zero median criteria used in Jurafsky et al. (2009) because C-SVMs under the l-2 metric provide no sparse weight guarantees.

FEMALE FLIRT		MALE FLIRT	
O_backchannel	-0.0369	S_you	0.0279
S_appreciation	-0.0327	S_negemotion	0.0249
O_appreciation	-0.0281	S_we	0.0236
O_question	0.0265	S_anger	0.0190
O_avimin	-0.0249	S_sexual	0.0184
S_turns	-0.0247	O_negemotion	0.0180
S_backchannel	-0.0245	O_avpmax	0.0174
O_you	0.0239	O_swear	0.0172
S_avtndur	0.0229	O_laugh	0.0164
S_avpmin	-0.0227	O_wordcount	0.0151
O_rate	0.0212	S_laugh	0.0144
S_laugh	0.0204	S_rate	0.0143
S_wordcount	0.0192	S_well	0.0131
S_well	0.0192	S_question	0.0131
O_negemotion	0.019	O_sexual	0.0128
S_repair_q	0.0188	S_completion	0.0128
O_sexual	0.0176	S_avpmax	0.011
O_overlap	-0.0176	O_completion	0.010
O_sdpmean	0.0171	O_sdimin	0.010
O_avimax	-0.0151	O_metatalk	-0.012
S_avpmean	-0.015	S_sdpsd	-0.015
S_question	-0.0146	S_avimin	-0.015
O_sdimin	0.0136	S_backchannel	-0.022
S_avpmax	0.0131		
S_we	-0.013		
S_I	0.0117		
S_assent	0.0114		
S_metatalk	-0.0107		
S_sexual	0.0105		
S_avimin	-0.0104		
O_uh	-0.0102		

Table 5: Feature weights (mean weights of the randomized runs) for the predictors with $|weight| > 0.01$ for the male and female classifiers. An S prefix indicates features of the speaker (the candidate flirter) while an O prefix indicates features of the other. Weights for autoencoder features were also significant but are omitted for compactness.

Features of the alter (the woman) that helped our system detect men who say they are flirting include the woman's laughing, sexual words or swear words, talking more, and having a higher f0 (max).

• Women who say they are flirting have a much expanded pitch range (lower pitch min, higher pitch max), laugh more, use more *I* and *well*, use repair questions but not other kinds of questions, use more sexual terms, use far less appreciations and backchannels, and use fewer, longer turns, with more words in general. Features of the alter (the man) that helped our system detect women who say they are flirting include the male use of *you*, questions, and faster and quieter speech.

We also summarize here the features for the perception classification task; predicting which people will be labeled by their dates as flirting. Here the task is the same as for Jurafsky et al. (2009) and the values are similar.

• Men who are labeled by their female date as flirting present many of the same linguistic behaviors as when they express their intention to flirt. Some of the largest differences are that men are perceived to flirt when they use less appreciations and overlap less, while these features were not significant for men who said they were flirting. We also found that fast speech and more questions are more important features for flirtation perception than intention.

• Women who are labeled by their male date as flirting also present much of the same linguistic behavior as women who intend to flirt. Laughter, repair questions, and taking fewer, longer turns were not predictors of women labeled as flirting, although these were strong predictors of women intending to flirt.

Both genders convey intended flirtation by laughing more, speaking faster, and using higher pitch. However, we do find gender differences; men ask more questions when they say they are flirting, women ask fewer, although they do use more repair questions, which men do not. Women use more "I" and less "we"; men use more "we" and "you". Men labeled as flirting are softer, but women labeled as flirting are not. Women flirting use much fewer appreciations; appreciations were not a significant factor in men flirting.

8 Human Performance on this task

To evaluate the performance of our classifiers we compare against human labeled data.

We used the same test set as for our machine classifier; recall that this was created by taking the top ten percent of Likert ratings of the speaker's intention ratings by gender and called those positive for flirtation intention. We constructed negative examples by taking the bottom ten percent of intention Likert ratings. We called the interlocutor correct on the positive examples if the interlocutor's rating was greater than 5. Symmetrically for the negative examples, we said the interlocutor was correct if their rating was less than or equal to 5. Note that this metric is biased somewhat toward the humans and against our systems, because we do not penalize for intermediate values, while the system is trained to make binary predictions only on extremes. The results of the human perceivers on classifying flirtation intent are shown in Table 6.

Male speaker (Female perceiver)	Female speaker (Male perceiver)
62.2%	56.2%

Table 6: Accuracy of human listeners at labeling speakers as flirting or not.

We were quite surprised by the poor quality of the human results. Our system outperforms both men's performance in detecting women flirters (system 71.5% versus human 56.2%) and also women's performance in detecting male flirters (system 69.0% versus human 62.2%).

Why are humans worse than machines at detecting flirtation? We found a key insight by examining how the participants in a date label themselves and each other. Table 7 shows the 1-10 Likert values for the two participants in one of the dates, between Male 101 and Female 127. The two participants clearly had very different perspectives on the date. More important, however, we see that each participant labels their own flirting (almost) identically with their partner's flirting.

	I am flirting	Other is flirting
Male 101 says:	8	7
Female 127 says:	1	1

Table 7: Likert scores for the date between Female 127 and Male 101.

We therefore asked whether speakers in general tend to assign similar values to their own flirting and their partner's flirting. The Pearson correlation coefficient between these two variables (my perception of my own flirting, and my perception of other's flirting) is .73. By contrast, the poor performance of subjects at detecting flirting in their partners is coherent with the lower (.15) correlation coefficient between those two variables (my perception of the other's flirting, and the other's perception of their own flirting). This discrepancy is summarized in boldface in Table 8.

Since the speed-date data was also labeled for three other variables, we then asked the same question about these variables. As Table 8 shows, for all four styles, speakers' perception of others is strongly correlated with the speakers' perception of themselves, far more so than with what the others actually think they are doing.[3]

[3]This was true no matter how the correlations were run, whether with raw Likert values, with ego-centered (transformed) values and with self ego-centered but other raw.

Variable	Self-perceive-Other & Self-perceive-Self	Self-perceive-Other & Other-perceive-Other
Flirting	**.73**	**.15**
Friendly	.77	.05
Awkward	.58	.07
Assertive	.58	.09

Table 8: Correlations between speaker intentions and perception for all four styles.

Note that although perception of the other does not correlate highly with the other's intent for any of the styles, the correlations are somewhat better (.15) for flirting, perhaps because in the speed-date setting speakers are focusing more on detecting this behavior (Higgins and Bargh, 1987). It is also possible that for styles with positive valence (friendliness and flirting) speakers see more similarity between the self and the other than for negative styles (awkward and assertive) (Krahé, 1983).

Why should this strong bias exist to link self-flirting with perception of the other? One possibility is that speakers are just not very good at capturing the intentions of others in four minutes. Speakers instead base their judgments on their own behavior or intentions, perhaps because of a bias to maintain consistency in attitudes and relations (Festinger, 1957; Taylor, 1970) or to assume there is reciprocation in interpersonal perceptions (Kenny, 1998).

9 Conclusion

We have presented a new system that is able to predict flirtation intention better than humans can, despite humans having access to vastly richer information (visual features, gesture, etc.). This system facilitates the analysis of human perception and human interaction and provides a framework for understanding why humans perform so poorly on intention prediction.

At the heart of our system is a core set of prosodic, dialogue, and lexical features that allow for accurate prediction of both flirtation intention and flirtation perception. Since previous word lists don't capture sufficient lexical information, we used an autoencoder to automatically capture new lexical cues. The autoencoder shows potential for being a promising feature extraction method for social tasks where cues are domain specific.

Acknowledgments: Thanks to the anonymous reviewers and to a Google Research Award for partial funding.

References

J. Ang, R. Dhillon, A. Krupski, E. Shriberg, and A. Stolcke. 2002. Prosody-Based Automatic Detection of Annoyance and Frustration in Human-Computer Dialog. In *INTERSPEECH-02*.

Paul Boersma and David Weenink. 2005. Praat: doing phonetics by computer (version 4.3.14). [Computer program]. Retrieved May 26, 2005, from `http://www.praat.org/`.

M. A. Cohn, M. R. Mehl, and J. W. Pennebaker. 2004. Linguistic markers of psychological change surrounding September 11, 2001. *Psychological Science*, 15:687–693.

Leon Festinger. 1957. *A Theory of Cognitive Dissonance*. Row, Peterson, Evanston, IL.

E. Tory Higgins and John A. Bargh. 1987. Social cognition and social perception. *Annual Review of Psychology*, 38:369–425.

G. E. Hinton and R. R Salakhutdinov. 2006. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507.

G. E. Hinton, S. Osindero, and Y. Teh. 2006. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527–1554.

Dan Jurafsky, Rajesh Ranganath, and Dan McFarland. 2009. Extracting social meaning: Identifying interactional style in spoken conversation. In *NAACL HLT 2009*, Boulder, CO.

David Kenny. 1998. *Interpersonal Perception: A Social Relations Analysis*. Guilford Press, New York, NY.

B. Krahé. 1983. Self-serving biases in perceived similarity and causal attributions of other people's performance. *Social Psychology Quarterly*, 46:318–329.

C. M. Lee and Shrikanth S. Narayanan. 2002. Combining acoustic and language information for emotion recognition. In *ICSLP-02*, pages 873–876, Denver, CO.

Jackson Liscombe, Jennifer Venditti, and Julia Hirschberg. 2003. Classifying Subject Ratings of Emotional Speech Using Acoustic Features. In *INTERSPEECH-03*.

Anmol Madan, Ron Caneel, and Alex Pentland. 2005. Voices of attraction. Presented at Augmented Cognition, HCI 2005, Las Vegas.

François Mairesse and Marilyn Walker. 2008. Trainable generation of big-five personality styles through data-driven parameter estimation. In *ACL-08*, Columbus.

François Mairesse, Marilyn Walker, Matthias Mehl, and Roger Moore. 2007. Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of Artificial Intelligence Research (JAIR)*, 30:457–500.

Andrew Y. Ng. 2004. Feature selection, L1 vs. L2 regularization, and rotational invariance. In *ICML 2004*.

J. W. Pennebaker and T. C. Lay. 2002. Language use and personality during crises: Analyses of Mayor Rudolph Giuliani's press conferences. *Journal of Research in Personality*, 36:271–282.

J. W. Pennebaker, R.E. Booth, and M.E. Francis. 2007. Linguistic inquiry and word count: LIWC2007 operator's manual. Technical report, University of Texas.

Alex Pentland. 2005. Socially aware computation and communication. *Computer*, pages 63–70.

Irina Rish. 2001. An empirical study of the naive bayes classifier. In *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence*.

Andrew Rosenberg and Julia Hirschberg. 2005. Acoustic/prosodic and lexical correlates of charismatic speech. In *EUROSPEECH-05*, pages 513–516, Lisbon, Portugal.

S. S. Rude, E. M. Gortner, and J. W. Pennebaker. 2004. Language use of depressed and depression-vulnerable college students. *Cognition and Emotion*, 18:1121–1133.

B. Scholkopf, K.K. Sung, CJC Burges, F. Girosi, P. Niyogi, T. Poggio, and V. Vapnik. 1997. Comparing support vector machines with Gaussian kernels to radialbasis function classifiers. *IEEE Transactions on Signal Processing*, 45(11):2758–2765.

Howard Taylor. 1970. Chapter 2. In *Balance in Small Groups*. Von Nostrand Reinhold Company, New York, NY.