Naïve Bayes and Natural Language Processing

Abe Kazemzadeh

HackNight@Carbon5

08/08/2012

Outline

1. Introduction
2. Mathematical Formulation
3. Naïve Bayes in NLTK
4. Applications
5. Appendix

Acknowledgements

- Carbon 5.
- Labmates/advisor/coauthors.

Python for NLP

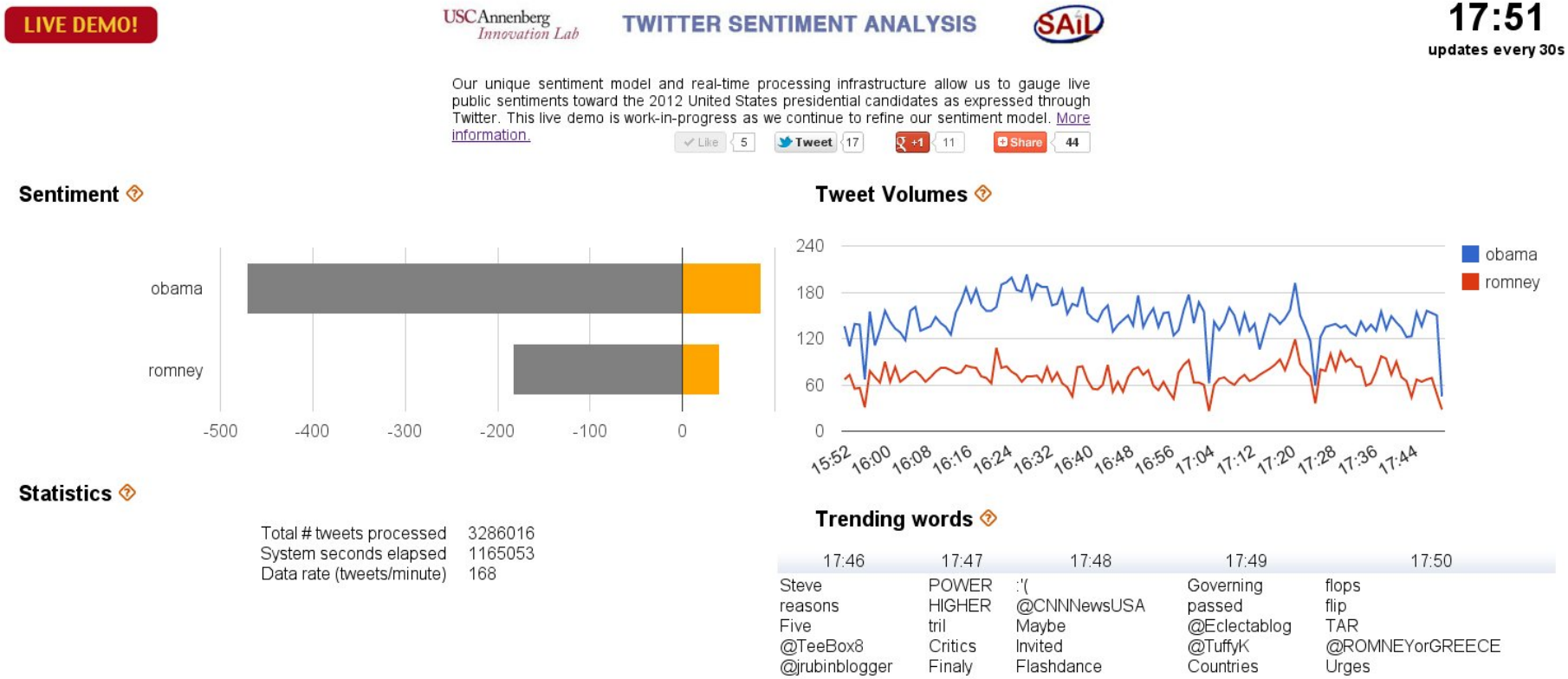
- I've seen a shift from Perl to Python as the predominant language for Natural Language Processing (NLP).
- A lot of NLP involves counting words for statistical estimation.
  - Perl is great for quick and dirty counting words.
  - Python is better for reuse and insightful program structure.
- NLTK, Natural Language Toolkit is the main python NLP library.
  - http://nltk.org
  - Offers many modules
    - Part of speech tagging, parsing, other types of machine learning...
  - We'll focus on classification and probability estimation using the Naïve Bayes class in the classifery module.

Applications

- We'll look at two applications:
  - Text classification
  - Dialog agent behavior

Text Classification/Sentiment Analysis

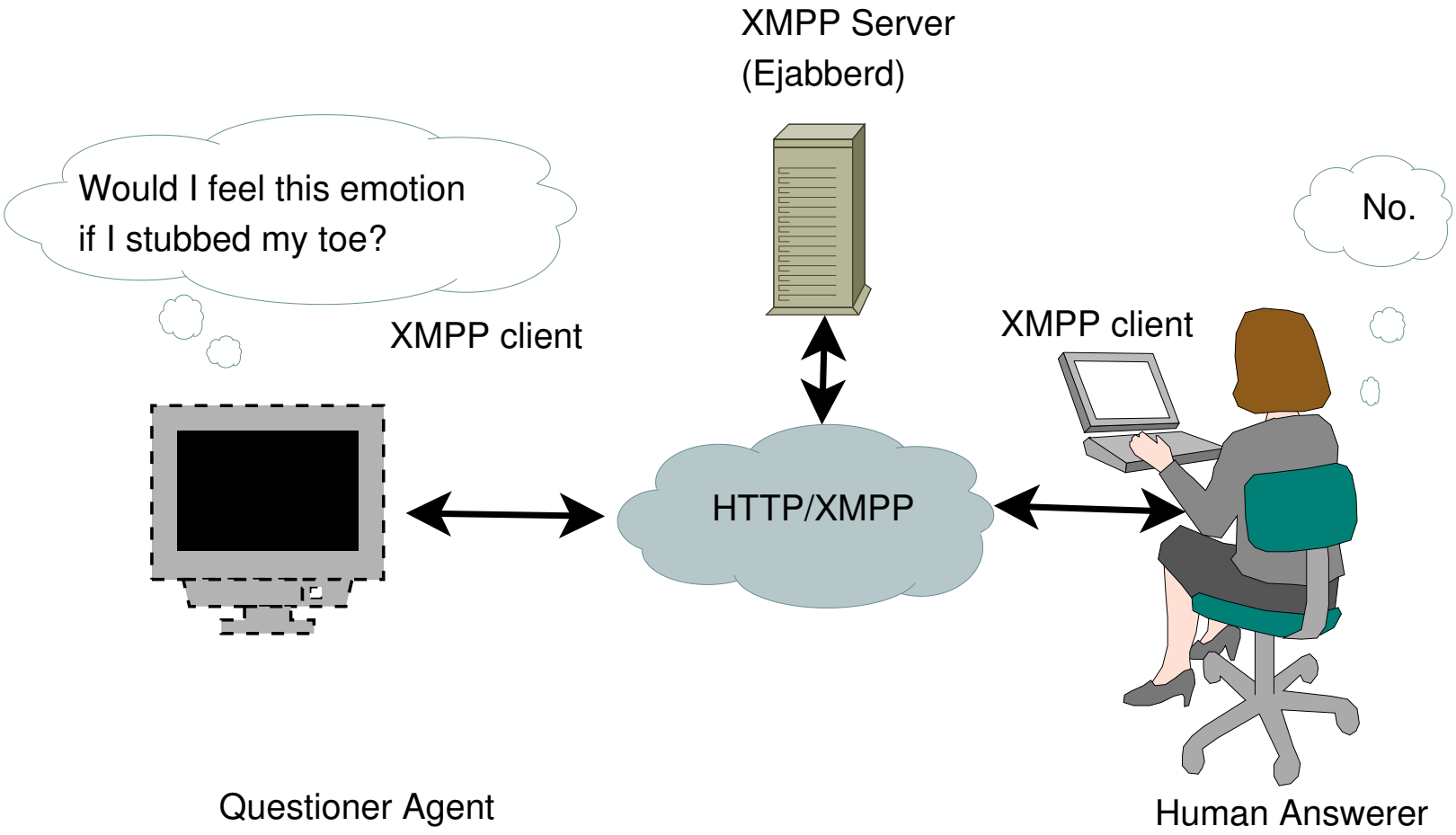
http://politics.twittersentiment.org





EMO20Q: Emulating Intelligent Behavior

Try the demo: http://ark.usc.edu/~abe/wsgi_questioner (can be slow)



Naïve Bayes



Thomas Bayes, 1701-1761.

$$posterior = \frac{likelihood * prior}{evidence}$$

$$P(model|data) = \frac{P(data|model) * P(model)}{P(data)}$$

- The naïvité/simplicity belies some deep insight and lots of academic debate.

Equations 1

$$posterior = \frac{likelihood * prior}{evidence}$$

$$P(model|data) = \frac{P(data|model) * P(model)}{P(data)}$$

- Prior
  - What model/explanation do we expect, before we observe anything?
- Likelihood
  - What is the probability of the observed data, given a particular model/explanation?
- Evidence
  - What is the probability of the observation, regardless of the model/explanation?
- Posterior
  - What model/explanation is the most likely given the data?

Example

Example:

the motorcade of Obama's helicopters that just flew over my house << seriously is that what I pay taxes for? #unnecessary take a fucking bus

- Prior
  - Political discussion on twitter tends to be more "negative" than discussion about movies
- Likelihood
  - The likelihood of seeing "fucking" when the model is "negative".
- Evidence
  - Probability of seing "motorcade" is less than the probability of "the"; the whole sentence probability can be seen as the product of the word probabilities.
- Posterior
  - This sentence is best explained by the "negative" model ($P(negative|tweet) > P(positive|tweet)$)

Equations 2

$$posterior = \frac{likelihood * prior}{evidence}$$

$$P(model|data) = \frac{P(data|model) * P(model)}{P(data)}$$

- Prior
  - P("positive") = count("positive")/N, ...
- Likelihood
  - P("the"|"positive") = count("the")/count("positive"), ...
- Evidence
  - P("the")=count("the")/W, or
    $P("the") = \sum_{m \in \{positive, negative, ...\}} P("the"|m)$
- Posterior
  - Plug in the rest and this is what you get.
    - Leave out P(data) if all you want is a classification or sorting

More Detail

$$P(C|w_1, ..., w_t) = \frac{\prod_{i=1}^{t} [P(w_i|C)] P(C)}{\prod_{i=1}^{t} P(w_i)}$$

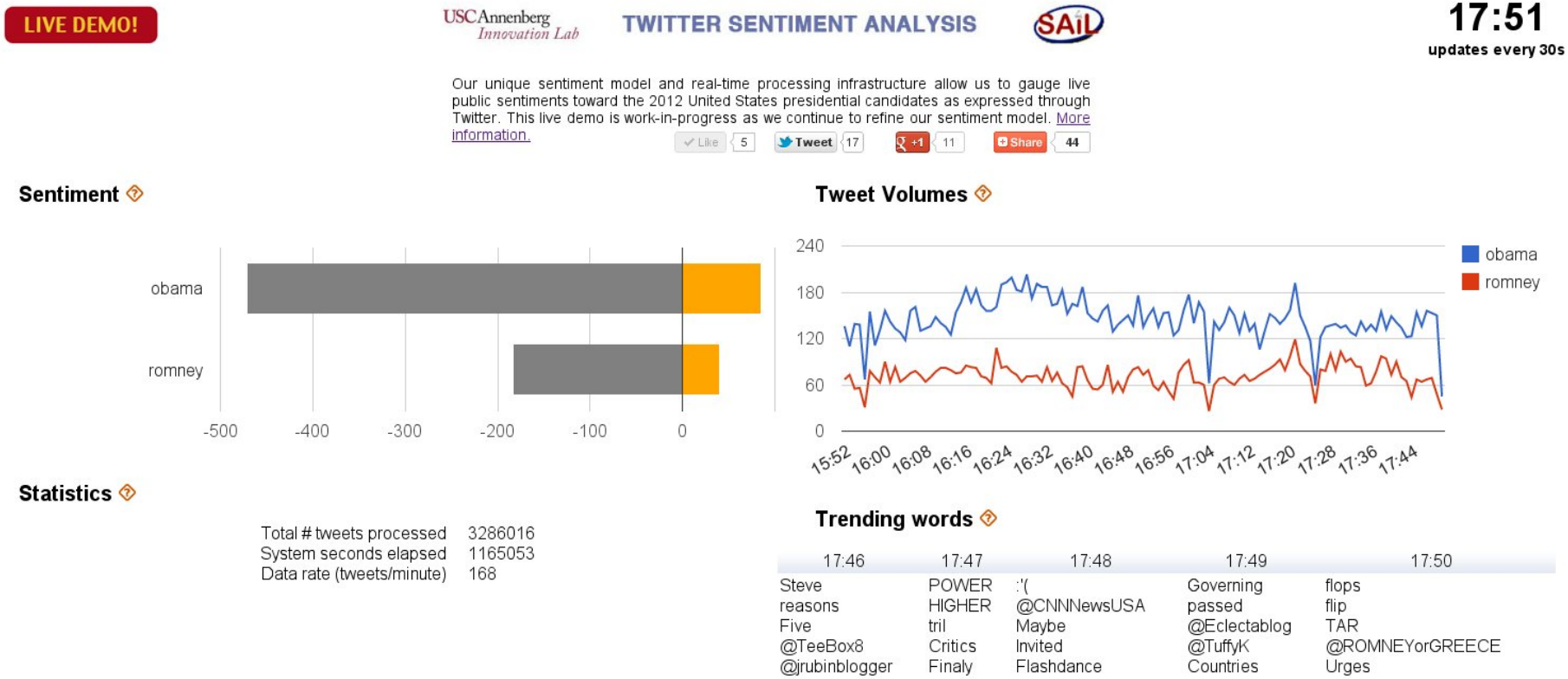
- Given a set of words $w_1 ... w_t$, what is the probability that it belongs to class/category C.

Class Structure

- Pull up `http://nltk.googlecode.com/svn/trunk/doc/api/nltk-module.html`
- Two modules will be relevant
  - `nltk.classify`
  - `nltk.probability`

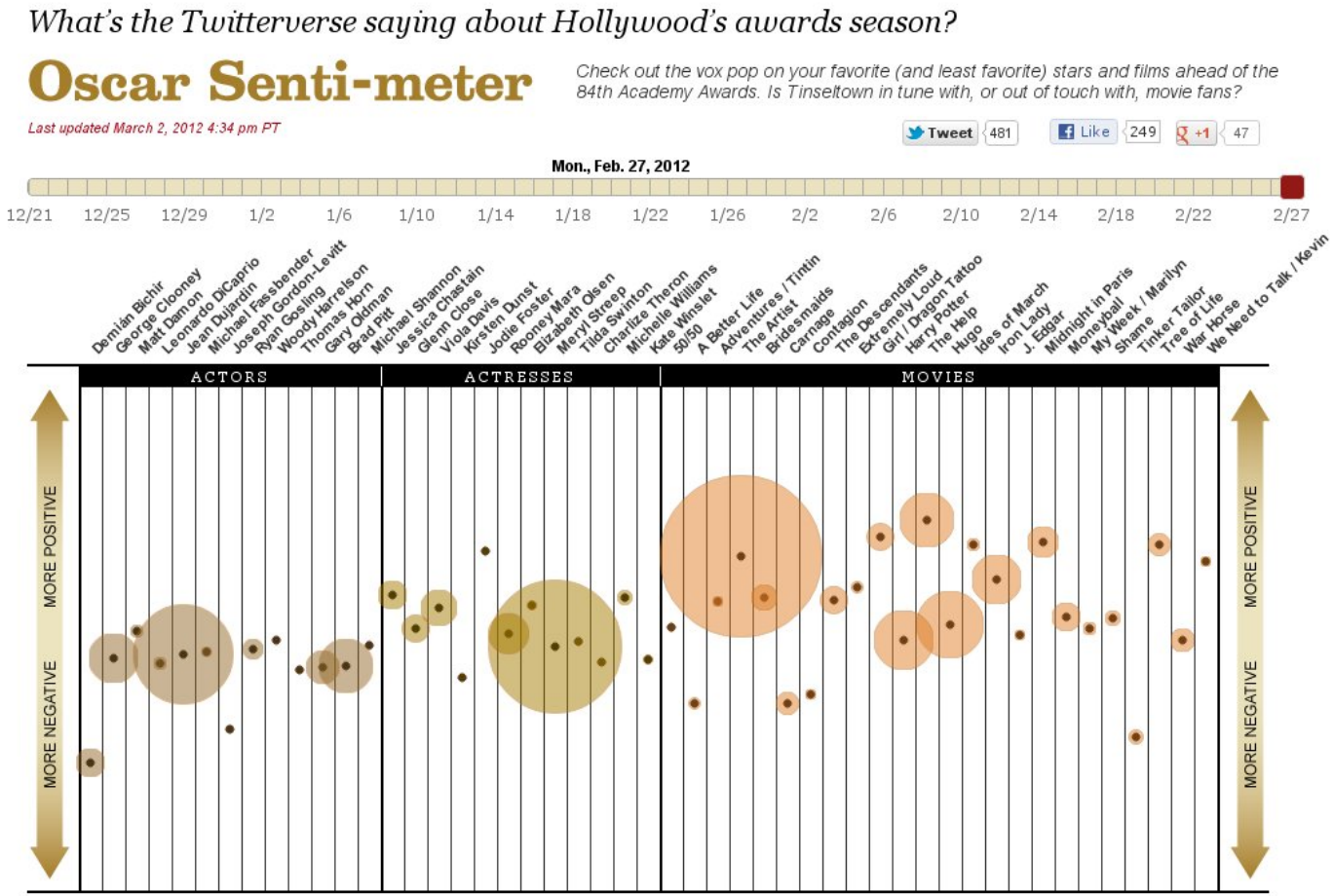
Text Analytics/Sentiment Analysis

http://politics.twittersentiment.org



Text Analytics/Sentiment Analysis

http://graphics.latimes.com/senti-meter/



Example Code: Get Features

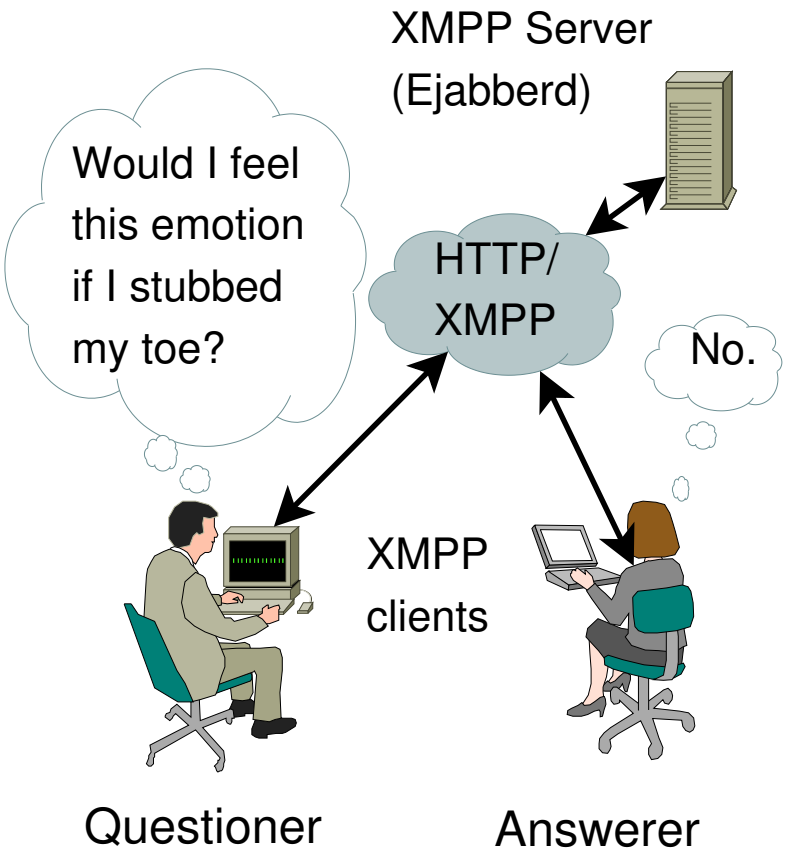
```
def features(tweet, n):
    feats = defaultdict(bool)
    words = ['<s>'] + tweet[u'tokens'] + ['</s>']
    for i in range(len(words)):
        for j in range(i + 1, i + n + 1):
            feat = " ".join(words[i:j])
            feats[feat] = True
    return feats

datafile = open("data.txt")
for line in datafile:
  line = line.decode("utf-8")
  text, label = line.strip().split("\t")
  data[text] += [(features(text, n), label)]
datafile.close()
# allocate train and test data, flattening list of lists
train = [y for x in data.values()[:13000] for y in x]
test = [x[0] for x in data.values()[13000:]]
```

Example Code: Train and Test

```
# train the model
classifier = nltk.NaiveBayesClassifier.train(train)
# classify and write hypothesis labels
hypothesis = open("results/hyp.txt", "w")
print >> hypothesis, "\n".join(classifier.batch_classify([x[0] for x in test])
hypothesis.close()
# write reference labels
reference = open("results/ref.txt", "w")
print >> reference, "\n".join([x[1] for x in test])
reference.close()
```

EMO20Q: Crowd-Sourcing a Theory of Emotions



(*Kazemzadeh et al, ACII 2011*)

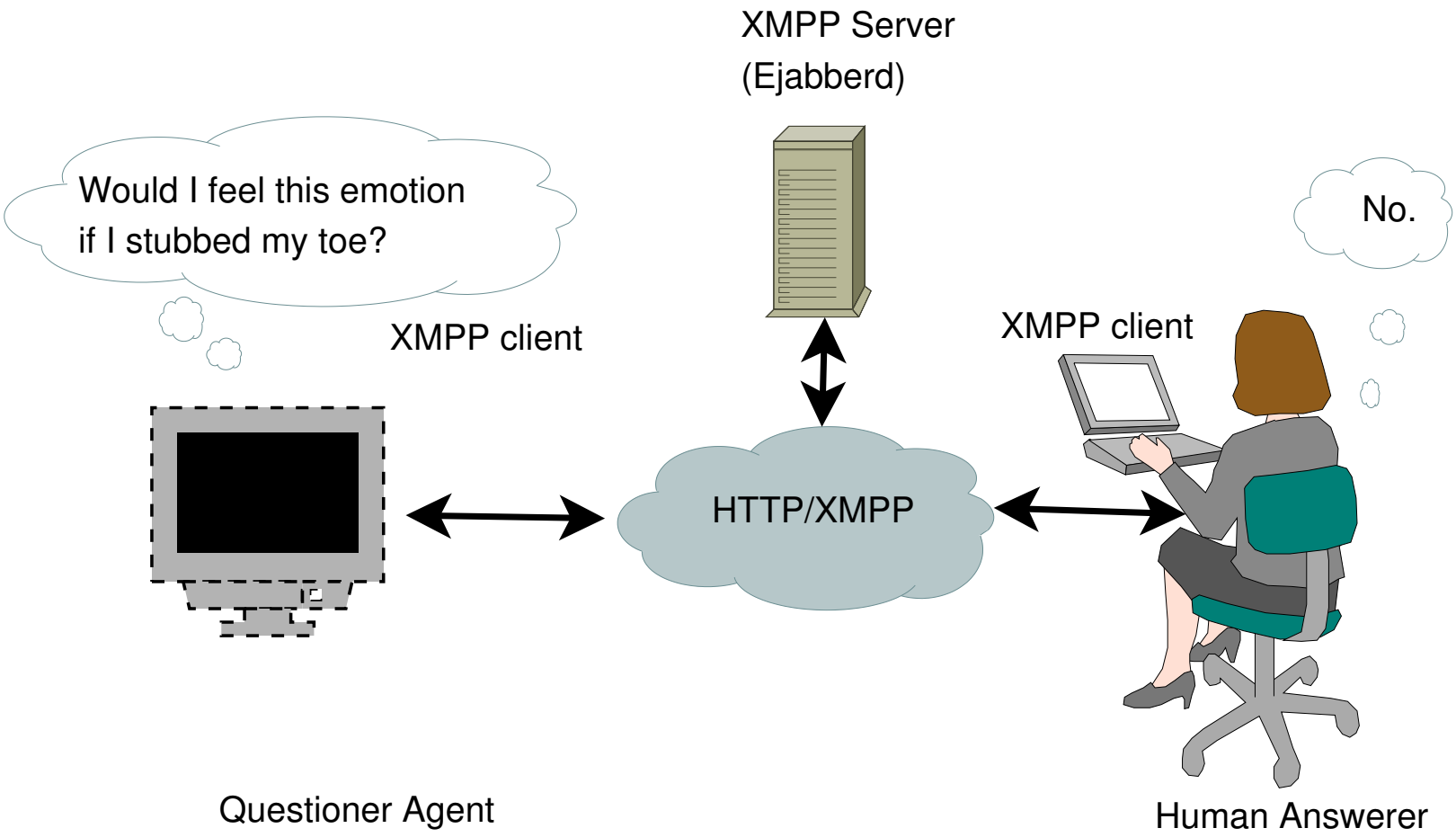
We use EMO20Q to

- See what people consider to be emotions
- How their opinions about emotions differ
- How often they come to agreement
- How long it takes to come to agreement

"A Shannon game for emotions"

Evaluation: Turing Test

`http://code.google.com/p/emotion-twenty-questions/`



Computational Model For Questioner

- Use each question as a feature.
- Assume that the feature values are “yes”, “no”, “other”, or “missing”.
- At each turn, pick a question that will best narrow down the set of possible emotions, based on the human-human data.
- Sequential Bayesian update algorithm uses Naive Bayes but defers classification.
  - Start with a uniform distribution over all emotion words, then update the probabilities as each question is answered.
- Based on a human-computer user study (partially conducted at Carbon 5), the system correctly guessed 44% of the time (the test set included out of vocabulary words).

Sequential Bayesian Belief Update

$$P(\varepsilon|f_1,...,f_t) = \frac{\prod_{i=1}^{t}[P(f_i|\varepsilon)]\,P(\varepsilon)}{\prod_{i=1}^{t}P(f_i)}. \tag{1}$$

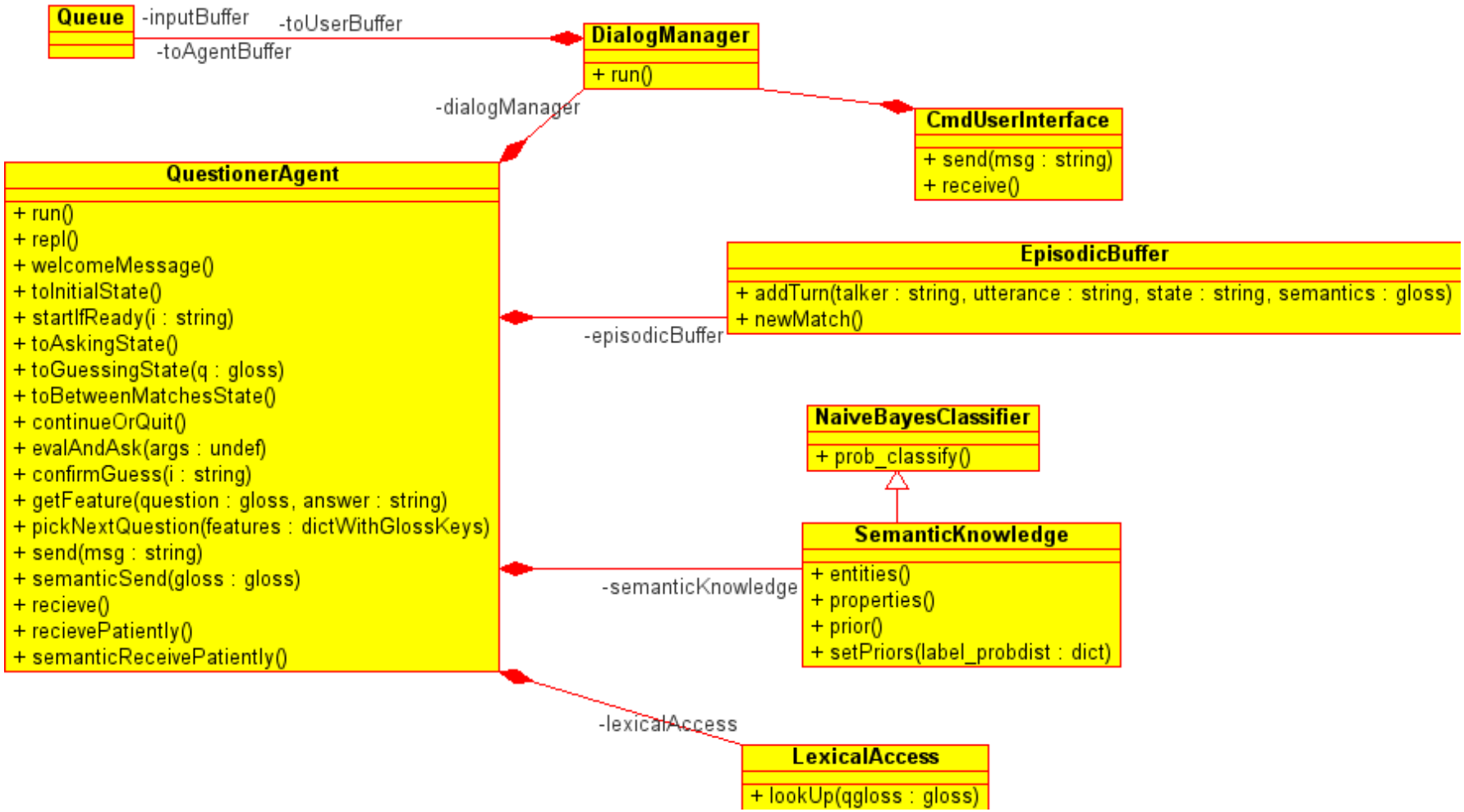
$$P(\varepsilon|f_1,...,f_t) = \frac{P(f_t|\varepsilon)P(\varepsilon|f_1,...,f_{t-1})}{P(f_t)}$$

$$\begin{aligned}
\beta_{0,k} &= P(\varepsilon = e_k) = 1/|E| \\
\beta_{1,k} &= \frac{P(f_1|\varepsilon = e_k)}{P(f_1)}\beta_{0,k} \\
\beta_{t,k} &= \frac{P(f_t|\varepsilon = e_k)}{P(f_t)}\beta_{t-1,k}
\end{aligned} \tag{2}$$

Algorithm

Input: $F = Q \times A$, E, and $P(f|\varepsilon)$
$\beta_{0,k} \leftarrow 1/|E|$, $\forall k = 1...|E|$
for $i = 1$ **to** d **do**
  $q^{(i)} = \underset{q \in Q}{\text{argmax}} P((q, \text{'yes'})|\varepsilon)$
  Print $q^{(i)}$
  $a^{(i)} \leftarrow$ user's input answer
  $f_i \leftarrow (q^{(i)}, a^{(i)})$
  $\beta_{i,k} \leftarrow \beta_{i-1,k} \cdot P(f_i|\varepsilon = e_k)/P(f_i)$, $\forall k = 1...|E|$
  if ($q^{(i)}$ is identity question for $e_k \wedge a^{(i)} =$ 'yes') **then**
    Return: $e^* = e_k$
  end if
  if ($q^{(i)}$ is identity question for $e_k \wedge a^{(i)} =$ 'no') **then**
    $\beta_{i,k} \leftarrow 0$
  end if
end for
$k^* \leftarrow \underset{k \in 1...|E|}{\text{argmax}} [\beta_{i,k}]$
$e^* \leftarrow e_{k^*}$
Return: most likely emotion given observations: e^*

Class Diagram



Results

Table: Experimental results.

difficulty	% success	avg. turns	% in vocab.	naturalness
easy	73%	11.4	100%	6.9
medium	46%	17.3	93%	5.5
difficult	13%	18.2	60%	5.8
total	44%	15.6	84%	6.1

Table: Observed emotion words by difficulty.

difficulty	examples
easy	happiness, anger, sadness, calm, confusion
medium	anger, confusion, contentment, curiosity, depression, disgust, excitement, fear, hate, irritation, love, melancholy, sorrow, surprise, tiredness
difficult	devastation, disgust, ecstasy, ennui, frustration, guilt, hope, irritation, jealousy, morose, proud, remorse, vindication, zealousness

Naïve Bayes and Natural Language Processing

Abe Kazemzadeh

HackNight@Carbon5

08/08/2012

General Characteristics of a NL Theory

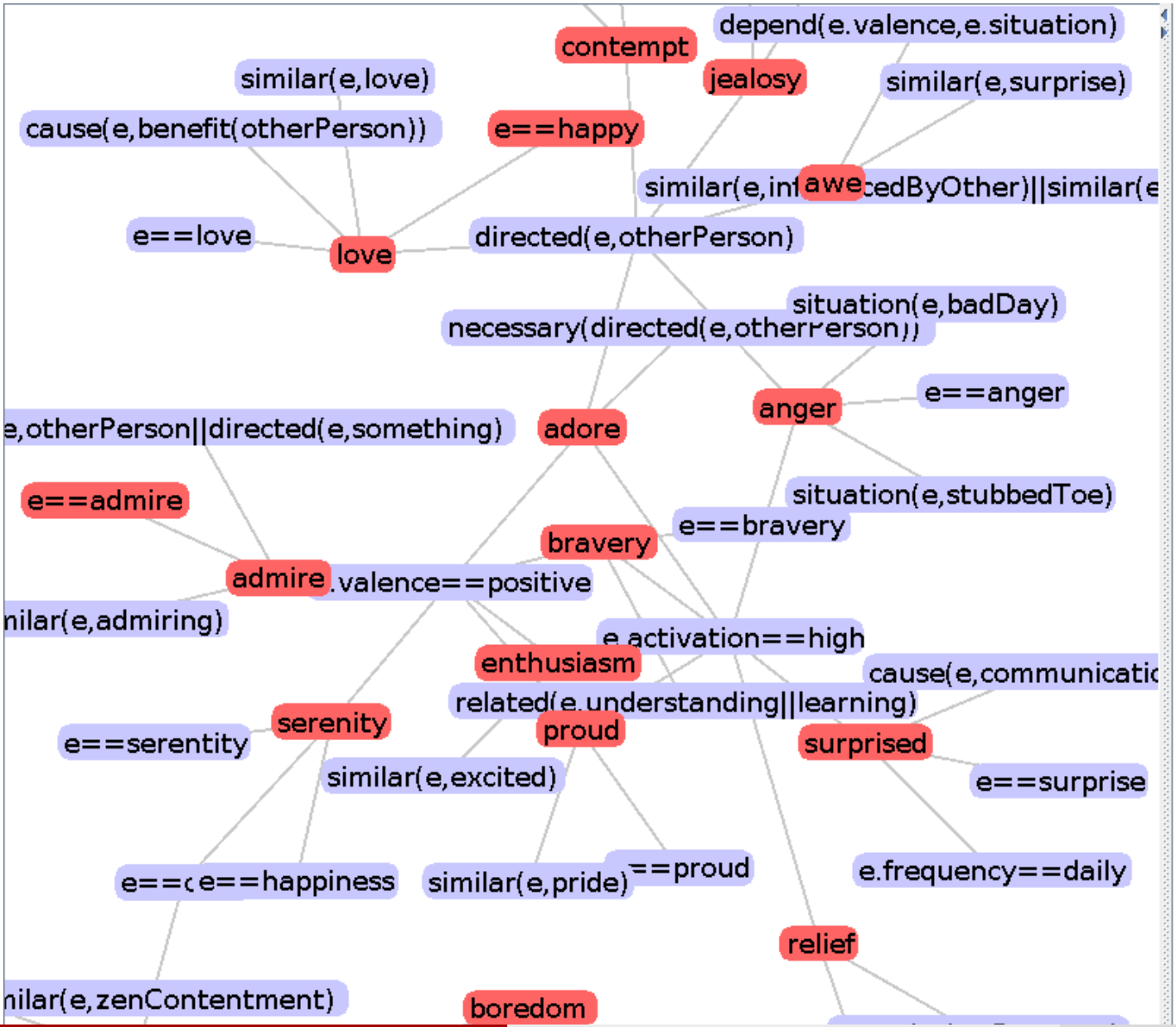
Question Category	**Examples**
identity (42%)	*is it angry?*
	guilt?
attribute (13%)	*is it something one feels for long periods of time?*
	is it a strong emotion?
similarity/ subsethood (10%)	*is the emotion a type of or related to content or zen contentment (is that a word?_)*
	so it's similar to excited?
situational (14%)	*is the emotion more likely to occur when you are tired?*
	would i feel this if my dog died?
behavior (3%)	*you can express it in an obvious way by sighing?*
	do adults usually try to discourage children from feeling this?
causal (7%)	*yes. mynext question is can it harm anyone besides the feeler?*
	I think I know, but I'll ask one more question...does it ever cause children to wake up and cry?
social (8%)	*are you less likely to experience the emotion when around good firiends?*
	13)would you feel that towards someone who is superior to you?
miscelaneous (3%)	*i dont' know if this is a valid question, but does it start with the letter D?*
	or an aspirational emotion?
	does the word function or can be conjugated as anything eles? i.e. can it be a verb too?

EMO20Q Results: Human-Human

- 110 matches from 25 players.
- 85% of matches terminated successfully
  - c.f., annotator agreement for emotional data annotation: 67-83% with 9 labels.
- The mean and median number of questions was 12 and 10, respectively.
- In the data, a total of 71 unique emotions words.
- 174 emotion words considering words that were referred to in questions.
- 1228 questions-asking events.

(Kazemzadeh2 et al. *under review)*

EMO20Q Graph



Dealing with Sparsity: Standardization

Table: Examples of question standardization.

Standardized Question	**Examples**
cause(emptySet,e)	*can you feel the emotion without any external events that cause it?* *is it an emotion that just pops up spontaneously (vs being triggered by something)?*
cause(otherPerson,e)	*is it caused by the person that it's directed at?* *Do you need someone to pull this emotion out of you or evoke it? if so, who is it?*
e.valence==negative	*is it considered a negative thing to feel?* *2) so is it a negative emotion?*
situation(e,birthday)	*would you feel this if it was your birthday?* *is it a socially acceptable emotion, say, at a birthday party?*
e==frustration	*oh, is it frustrated?* *frustration?*

EMO20Q Algorithm

Input: $F = Q \times A$, E, and $P(f|\varepsilon)$
$\beta_{0,k} \leftarrow 1/|E|, \forall k = 1...|E|$
for $i = 1$ **to** d **do**
  $q^{(i)} = \underset{q \in Q}{\text{argmax}} P((q, \text{'yes'})|\varepsilon)$
  Print $q^{(i)}$
  $a^{(i)} \leftarrow$ user's input answer
  $f_i \leftarrow (q^{(i)}, a^{(i)})$
  $\beta_{i,k} \leftarrow \beta_{i-1,k} \cdot P(f_i|\varepsilon = e_k)/P(f_i), \forall k = 1...|E|$
  if ($q^{(i)}$ is identity question for $e_k \wedge a^{(i)} =$ 'yes') **then**
    Return: $e^* = e_k$
  end if
  if ($q^{(i)}$ is identity question for $e_k \wedge a^{(i)} =$ 'no') **then**
    $\beta_{i,k} \leftarrow 0$
  end if
end for
$k^* \leftarrow \underset{k \in 1...|E|}{\text{argmax}} [\beta_{i,k}]$
$e^* \leftarrow e_{k^*}$
Return: most likely emotion given observations: e^*

Issues

- "Concepts" vs. holophrastic indeterminacy.
  - Need to balance philosophical concerns with practical issues, e.g., theoretical object ≈virtual objects so it's a useful abstraction.
- Hypostatic abstraction: happy vs. happiness.