Modeling Natural Text

David Kauchak
CS458
Fall 2012

Admin

- Final project
  - Paper draft
    - due next Friday by midnight
  - Saturday, I'll e-mail out 1-2 paper drafts for you to read
  - Send me your reviews by Sunday at midnight
  - Monday morning, I'll forward these so you can integrate comments
  - Initial code submission
    - Make sure to start integrating your code sooner than later
    - Initial code submission due next Friday

- Final project continued
  - At the beginning of class on Tuesday and Thursday we'll spend 15 min. discussing where things are at
  - Any support from me?
    - let me know sooner than later...

Watson paper discussion

- First application attempts

- How did the discussion go?
- One more paper discussion next Tuesday...
Modeling natural text

Questions:
- What are the key topics in the text?
- What is the sentiment of the text?
- Who/what does the article refer to?
- What are the key phrases?

Phenomena:
- Synonymy
- Sarcasm/hyperbole
- Variety of language (slang), misspellings, coreference (e.g., pronouns like he/she)

Applications

- Search engines
- Search
- Advertising
- Corporate databases
- Language generation
- Speech recognition
- Machine translation
- Text simplification
- Text classification and clustering
- SPAM
- Yahoo!
- Document hierarchies
- Sentiment analysis

Document modeling:
Learn a probabilistic model of documents

Predict the likelihood that an unseen document belongs to a set of documents

Model should capture text characteristics

Training a document model

?
Applying a document model

Document model: what is the probability the new document is in the same “set” as the training documents?

Applications?

Application: text classification

Category
- sports
- politics
- entertainment
- business
- ...

Sentiment
- spam
- not-spam
- positive
- negative

Text classification: Training

SPAM
non-SPAM

model parameter estimation

Text classification: Applying

Is it SPAM or non-SPAM?

probability of document being SPAM

which is larger?

probability of document being non-SPAM
Representation and Notation

Standard representation: bag of words
- Fixed vocabulary ~50K words
- Documents represented by a count vector, where each dimension represents the frequency of a word

Clinton said banana repeatedly last week on tv, "banana, banana, banana"

Representation allows us to generalize across documents

Downside?

Word burstiness

What is the probability that a political document contains the word "Clinton" exactly once?

The Stacy Koon-Lawrence Powell defense! The decisions of Janet Reno and Bill Clinton in this affair are essentially the moral equivalents of Stacy Koon’s. ...

\[ p(\text{"Clinton"}=1|\text{political}) = 0.12 \]

What is the probability that a political document contains the word "Clinton" exactly twice?

The Stacy Koon-Lawrence Powell defense! The decisions of Janet Reno and Bill Clinton in this affair are essentially the moral equivalents of Stacy Koon’s. Reno and Clinton have the advantage in that they investigate themselves.

\[ p(\text{"Clinton"}=2|\text{political}) = 0.05 \]
Word burstiness in models

\[ p("\text{Clinton}" = 2 | \text{political}) = 0.05 \]

Many models incorrectly predict:

\[ p("\text{Clinton}" = 2 | \text{political}) \approx p("\text{Clinton}" = 1 | \text{political})^2 \]

\[ 0.05 \neq 0.0144 \text{ (0.12)}^2 \]

And in general, predict:

\[ p("\text{Clinton}" = i | \text{political}) \approx p("\text{Clinton}" = 1 | \text{political})^i \]

Word count probabilities

common words – 71% of word occurrences and 1% of the vocabulary
average words – 21% of word occurrences and 10% of the vocabulary
rare words – 8% of word occurrences and 89% of the vocabulary

The models...
Multinomial model

20 rolls of a fair, 6-side die - each number is equally probable

(1, 10, 5, 1, 2, 1) (3, 3, 3, 3, 4, 4)

Which is more probable?

Multinomial model for text

Many more "sides" on the die than 6, but the same concept...

(4, 1, 1, 0, 1, 0, 0, ...)
Generative Story

To apply a model, we're given a document and we obtain the probability.

We can also ask how a given model would generate a document.

This is the “generative story” for a model.

Multinomial Urn: Drawing words from a multinomial

Selected:

Drawing words from a multinomial

Selected: $w_6$

Put a copy of $w_1$ back.
Drawing words from a multinomial

Selected: $w_1, w_3$

Put a copy of $w_1$ back

Put a copy of $w_2$ back

Selected: $w_1, w_2, w_3$
Drawing words from a multinomial

Does the multinomial model capture burstiness?

\[ p(\text{word}) \text{ remains constant, independent of which words have already been drawn (in particular, how many of this particular word have been drawn).} \]

**Multinomial probability simplex**

Generate documents containing 100 words from a multinomial with just 3 possible words

\[
\begin{align*}
\text{word 1} & \quad \text{word 2} & \quad \text{word 3} \\
0.31, & \quad 0.44, & \quad 0.25)
\end{align*}
\]
Multinomial word count probabilities

Multinomial does not model burstiness of average and rare words

Better model of burstiness: DCM
Dirichlet Compound Multinomial

Polya Urn process
- **KEY**: Urn distribution changes based on previous words drawn
- Generative story:
  - Repeat until document length hit
    - Randomly draw a word from urn – call it \( w_j \)
    - Put 2 copies of \( w_j \) back in urn

Drawing words from a Polya urn

Selected:
Drawing words from a Polya urn

Selected: \(w_1\)

Put 2 copies of \(w_1\), back

Adjust parameters

Drawing words from a Polya urn

Selected: \(w_2\)

Drawing words from a Polya urn

Selected: \(w_1\)

Drawing words from a Polya urn

Selected: \(w_0\), \(w_3\)

Put 2 copies of \(w_1\), back

Adjust parameters
Polya urn

Words already drawn are more likely to be seen again

Results in the DCM distribution

We can modulate burstiness by increasing/decreasing the number of words in the urn while keeping distribution the same
Controlling burstiness

Same distribution of words

Which is more bursty?

more bursty

less bursty

Burstiness with DCM

Multinomial

DCM

Down scaled (.31, .44, .25)

Medium scaled (.93, 1.32, .75)

Up scaled (2.81, 3.94, 2.25)

DCM word count probabilities

Reminder...
Modeling burstiness in other applications

Which model would be better: multinomial, DCM, other?

- User movie watching data
- Bags of M&M's
- Daily Flight delays

A look at the code… multinomial model

Training

```matlab
for i = 1:length(vectors)
    thetas(i,:) = log(sum(vector,1) + ones(1,size(vector,2))) -
    log(sum(sum(vector)) + size(vector,2));
```

Applying model

```matlab
for i = 1:length(vectors)
    probs = thetas(:,idx) * vectors{i}(:,idx)';
    [temp, decisions{i}] = max(probs);
```

DCM model

$$p(x | \alpha) = \int \frac{|x|!}{\prod_{i=1}^{r} x_i!} \left( \frac{x}{\prod_{i=1}^{r} \alpha_i} \right)^{\sum_{i=1}^{r} \alpha_i} \frac{\prod_{i=1}^{r} \Gamma(\nu_i) \prod_{i=1}^{r} \Gamma(\alpha_i)} {\prod_{i=1}^{r} \Gamma(\nu_i + \alpha_i) \prod_{i=1}^{r} \Gamma(\alpha_i)} \, d\theta$$

Experiments

How can we test different models quantitatively?
Experiments

Modeling one class: document modeling

Modeling alternative classes: classification

Two standard data sets

Industry sector (web pages)
- More classes
- Less documents per class
- Longer documents

20 newsgroups (newsgroup posts)
- Fewer classes
- More documents per class
- Shorter documents

Modeling a single class:
the fruit bowl

Student 1

Student 2

Goal: predict what the fruit mix will be for the following Monday (assign probabilities to options)

Modeling a single class/group

How well does a model predict unseen data?

Model 1

Model 2

Monday

Which model is better?

How would you quantify how much better?
Modeling evaluation: perplexity

Perplexity is the average of the negative log of the model probabilities on test data.

Model 1

Model 2

Use the same idea to measure the performance of the different models for modeling one set of documents.

Perplexity results

20 newsgroups data set

<table>
<thead>
<tr>
<th>Model</th>
<th>Perplexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multinomial</td>
<td>92.1</td>
</tr>
<tr>
<td>DCM</td>
<td>58.7</td>
</tr>
</tbody>
</table>

*Lower is better*

Ideally the model would have a perplexity of 0!

Significant increase in modeling performance!

Classification results

Precision = number correct / number of documents

<table>
<thead>
<tr>
<th></th>
<th>Industry</th>
<th>20 Newsgroups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multinomial</td>
<td>0.600</td>
<td>0.853</td>
</tr>
<tr>
<td>DCM</td>
<td>0.806</td>
<td>0.890</td>
</tr>
</tbody>
</table>

(results are on par with state of the art discriminative approaches!)

Next steps in text modeling

- Modeling textual phenomena like burstiness in text is important
- Better grounded models like DCM also perform better in applications (e.g., classification)

**Better models**

- text substitutability
- relax bag of words constraint (model co-occurrence)
- hierarchical models
- handling short phrases (tweets, search queries)

**Applications of models**

- multi-class data modeling (e.g., clustering)
- text similarity
- language generation applications (speech recognition, translation, summarization)
- handling short phrases (tweets, search queries)