Advanced Multimedia

Text Classification
Tamara Berg
Announcements

• Matlab basics lab – Feb 2
• Matlab string processing lab – Feb 7
• Special Lecture – Feb 9
  – Wang Center Lecture Hall 1
  – Nicolas Maigret (art of failure)
The Dream

- It’d be great if machines could
  - Process our email (usefully)
  - Translate languages accurately
  - Help us manage, summarize, and aggregate information
  - Use speech as a UI (when needed)
  - Talk to us / listen to us

- But they can’t:
  - Language is complex, ambiguous, flexible, and subtle
  - Good solutions need linguistics and machine learning knowledge

- So:
What is NLP?

- **Fundamental goal:** deep understand of broad language
  - Not just string processing or keyword matching!

- **End systems that we want to build:**
  - Ambitious: speech recognition, machine translation, information extraction, dialog interfaces, question answering…
  - Modest: spelling correction, text categorization…
What is NLP?

- Fundamental goal: *deep* understand of *broad* language
  - Not just string processing or keyword matching!

- End systems that we want to build:
  - Ambitious: speech recognition, machine translation, information extraction, dialog interfaces, question answering...
  - Modest: spelling correction, text categorization...

Slide from Dan Klein
What does categorization/classification mean?
Example: Spam Filter

- Input: email
- Output: spam/ham

Dear Sir.

First, I must solicit your confidence in this transaction, this is by virtue of its nature as being utterly confidential and top secret. …

TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES FOR ONLY $99

Ok, I know this is blatantly OT but I'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use, I know it was working pre being stuck in the corner, but when I plugged it in, hit the power nothing happened.
Example: Spam Filter

- **Input:** email
- **Output:** spam/ham
- **Setup:**
  - Get a large collection of example emails, each labeled “spam” or “ham”
  - Note: someone has to hand label all this data!
  - Want to learn to predict labels of new, future emails

---

**Disapproved Example:**

Dear Sir.

First, I must solicit your confidence in this transaction, this is by virtue of its nature as being utterly confidential and top secret. …

TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES FOR ONLY $99

---

**Approved Example:**

Ok, I know this is blatantly OT but I'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use. I know it was working pre being stuck in the corner, but when I plugged it in, hit the power nothing happened.
Example: Spam Filter

- **Input:** email
- **Output:** spam/ham
- **Setup:**
  - Get a large collection of example emails, each labeled “spam” or “ham”
  - Note: someone has to hand label all this data!
  - Want to learn to predict labels of new, future emails

**Features:** The attributes used to make the ham / spam decision

---

**Correct Example:**

Dear Sir.

First, I must solicit your confidence in this transaction, this is by virtue of its nature as being utterly confidential and top secret. …

TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES FOR ONLY $99

---

**Incorrect Examples:**

Ok, lknow this is blatantly OT but l'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use, l know it was working pre being stuck in the corner, but when l plugged it in, hit the power nothing happened.

---

Slide from Dan Klein
Example: Spam Filter

- Input: email
- Output: spam/ham
- Setup:
  - Get a large collection of example emails, each labeled “spam” or “ham”
  - Note: someone has to hand label all this data!
  - Want to learn to predict labels of new, future emails

- Features: The attributes used to make the ham / spam decision
  - Words: FREE!
  - Text Patterns: $dd, CAPS
  - Non-text: SenderInContacts
  - ...

---

Dear Sir.

First, I must solicit your confidence in this transaction, this is by virtue of its nature as being utterly confidential and top secret. …

TO BE REMOVED FROM FUTURE MAILINGS, SIMPLY REPLY TO THIS MESSAGE AND PUT "REMOVE" IN THE SUBJECT.

99 MILLION EMAIL ADDRESSES FOR ONLY $99

Ok, I know this is blatantly OT but I'm beginning to go insane. Had an old Dell Dimension XPS sitting in the corner and decided to put it to use, I know it was working pre being stuck in the corner, but when I plugged it in, hit the power nothing happened.
Example: Digit Recognition

- **Input:** images / pixel grids
- **Output:** a digit 0-9
- **Setup:**
  - Get a large collection of example images, each labeled with a digit
  - Note: someone has to hand label all this data!
  - Want to learn to predict labels of new, future digit images

- **Features:** The attributes used to make the digit decision
  - Pixels: (6,8)=ON
  - Shape Patterns: NumComponents, AspectRatio, NumLoops
  - ...

http://yann.lecun.com/exdb/mnist/index.html
Other Classification Tasks

- In classification, we predict labels y (classes) for inputs x

- Examples:
  - Spam detection (input: document, classes: spam / ham)
  - OCR (input: images, classes: characters)
  - Medical diagnosis (input: symptoms, classes: diseases)
  - Automatic essay grader (input: document, classes: grades)
  - Fraud detection (input: account activity, classes: fraud / no fraud)
  - Customer service email routing
  - ... many more

- Classification is an important commercial technology!
Applications of text classification in IR

- Language identification (classes: English vs. French etc.)
- The automatic detection of spam pages (spam vs. nonspam, example: googel.org)
- The automatic detection of sexually explicit content (sexually explicit vs. not)
- Sentiment detection: is a movie or product review positive or negative (positive vs. negative)
- Topic-specific or vertical search – restrict search to a “vertical” like “related to health” (relevant to vertical vs. not)
- Machine-learned ranking function in ad hoc retrieval (relevant vs. nonrelevant)
- Semantic Web: Automatically add semantic tags for non-tagged text (e.g., for each paragraph: relevant to a vertical like health or not)
Classification methods: 1. Manual

- Manual classification was used by Yahoo in the beginning of the web. Also: ODP, PubMed
- Very accurate if job is done by experts
- Consistent when the problem size and team is small
- Manual classification is difficult and expensive to scale.
- → We need automatic methods for classification.
Classification methods: 2. Rule-based

- There are “IDE” type development environments for writing very complex rules efficiently. (e.g., Verity)
- Often: Boolean combinations (as in Google Alerts)
- Accuracy is very high if a rule has been carefully refined over time by a subject expert.
- Building and maintaining rule-based classification systems is expensive.
• Machine Learning - how to select a model on the basis of data / experience
  Learning parameters (e.g. probabilities)
  Learning structure (e.g. dependencies)
  Learning hidden concepts (e.g. clustering)
Important Concepts

- **Data**: labeled instances, e.g. emails marked spam/ham
  - Training set
  - Held out set
  - Test set

- **Features**: attribute-value pairs which characterize each x

- **Experimentation cycle**
  - Learn parameters (e.g. model probabilities) on training set
  - (Tune hyperparameters on held-out set)
  - Compute accuracy of test set
  - Very important: never “peek” at the test set!

- **Evaluation**
  - Accuracy: fraction of instances predicted correctly

- **Overfitting and generalization**
  - Want a classifier which does well on test data
  - Overfitting: fitting the training data very closely, but not generalizing well
Representing Documents
The paper bag is a remarkable contrivance. It serves us constantly and inconspicuously. It folds flat, yet opens into a structure that can stand open upon the table while we eat our sandwiches from it and chat with friends.

If we take the bag apart, we find it's made from a single paper cylinder. One end of the cylinder has been folded into a complex 3-dimensional pattern and finished off with a bit of paste. It would be, and once was, costly to make, because each fragile cylinder had to be folded manually into that hardy sack.
Document Vectors

• Represent document as a “bag of words”
Example

• Doc1 = “the quick brown fox jumped”
• Doc2 = “brown quick jumped fox the”
Example

• Doc1 = “the quick brown fox jumped”
• Doc2 = “brown quick jumped fox the”

Would a bag of words model represent these two documents differently?
Document Vectors

- Documents are represented as “bags of words”
- Represented as vectors when used computationally
  - Each vector holds a place for every term in the collection
  - Therefore, most vectors are sparse
Document Vectors

• Documents are represented as “bags of words”
• Represented as vectors when used computationally
  • Each vector holds a place for every term in the collection
  • Therefore, most vectors are sparse

Lexicon – the vocabulary set that you consider to be valid words in your documents.
  Usually stemmed (e.g. running->run)
Document Vectors:
One location for each word.

<table>
<thead>
<tr>
<th>nova</th>
<th>galaxy</th>
<th>heat</th>
<th>h’wood</th>
<th>film</th>
<th>role</th>
<th>diet</th>
<th>fur</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>5</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Document Vectors:
One location for each word.

A   nova  galaxy  heat  h’wood  film  role  diet  fur

10  5     3     |      |      |      |      |

“Nova” occurs 10 times in text A
“Galaxy” occurs 5 times in text A
“Heat” occurs 3 times in text A
(Blank means 0 occurrences.)
## Document Vectors

<table>
<thead>
<tr>
<th>Document ids</th>
<th>Nova</th>
<th>Galaxy</th>
<th>Heat</th>
<th>Hollywood</th>
<th>Film</th>
<th>Role</th>
<th>Diet</th>
<th>Fur</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>10</td>
<td>5</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>5</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td>10</td>
<td>8</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td>10</td>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>G</td>
<td>5</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>6</td>
<td>10</td>
<td>2</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td></td>
<td></td>
<td>7</td>
<td>5</td>
<td>1</td>
<td></td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

Slide from Mitch Marcus
Vector Space Model

- **Documents are represented as vectors in term space**
  - Terms are usually stems
  - Documents represented by vectors of terms
- **A vector distance measures similarity between documents**
  - Document similarity is based on length and direction of their vectors
  - Terms in a vector can be “weighted” in many ways
# Document Vectors

<table>
<thead>
<tr>
<th>Document ids</th>
<th>nova</th>
<th>galaxy</th>
<th>heat</th>
<th>h'wood</th>
<th>film</th>
<th>role</th>
<th>diet</th>
<th>fur</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>10</td>
<td>5</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>5</td>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td>10</td>
<td>8</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
<td>9</td>
<td>10</td>
<td>5</td>
<td></td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>F</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>5</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td>9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>6</td>
<td>10</td>
<td>2</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>I</td>
<td></td>
<td></td>
<td>7</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Comparing Documents
Similarity between documents

\[ A = \begin{bmatrix} 10 & 5 & 3 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \]
\[ G = \begin{bmatrix} 5 & 0 & 7 & 0 & 0 & 9 & 0 & 0 \end{bmatrix}; \]
\[ E = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 10 & 10 & 0 \end{bmatrix}; \]
Similarity between documents

\[ A = [10 \ 5 \ 3 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0] ; \]
\[ G = [ 5 \ 0 \ 7 \ 0 \ 0 \ 9 \ 0 \ 0 \ 0] ; \]
\[ E = [ 0 \ 0 \ 0 \ 0 \ 0 \ 10 \ 10 \ 0 \ 0] ; \]

Treat the vectors as binary = number of words in common.

\[ Sb(A,G) = ? \]
\[ Sb(A,E) = ? \]
\[ Sb(G,E) = ? \]

Which pair of documents are the most similar?
Similarity between documents

A = [10 5 3 0 0 0 0 0];
G = [5 0 7 0 0 9 0 0];
E = [0 0 0 0 0 10 10 0];

Sum of Squared Distances (SSD) = ∑(X_i - Y_i)^2

SSD(A,G) = ?
SSD(A,E) = ?
SSD(G,E) = ?
Similarity between documents

\[ A = [10\ 5\ 3\ 0\ 0\ 0\ 0\ 0] ; \]

\[ G = [5\ 0\ 7\ 0\ 0\ 9\ 0\ 0] ; \]

\[ E = [0\ 0\ 0\ 0\ 0\ 10\ 10\ 0] ; \]

Angle between vectors: \[ \cos(\theta) = \frac{a \cdot b}{\|a\|\|b\|} \]

Dot Product: \[ a \cdot b = \sum_{i=1}^{n} a_i b_i = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n \]

Length (Euclidean norm): \[ \|a\| = \sqrt{a_1^2 + a_2^2 + \cdots + a_n^2} \]
Some words give more information than others

- Does the fact that two documents both contain the word “the” tell us anything? How about “and”? 

No definitive list but might include things like: [http://www.dcs.gla.ac.uk/idom/ir_resources/linguistic_resources/stop_words](http://www.dcs.gla.ac.uk/idom/ir_resources/linguistic_resources/stop_words)
Some words give more information than others

• Does the fact that two documents both contain the word “the” tell us anything? How about “and”? Stop words (noise words): Words that are probably not useful for processing. Filtered out before natural language is applied.

• Other words can be more or less informative.

No definitive list but might include things like:
http://www.dcs.gla.ac.uk/idom/ir_resources/linguistic_utils/stop_words
Classifying Documents
Here the vector space is illustrated as having 2 dimensions. How many dimensions would the data actually live in?
Query document – which class should you label it with?
Classification by Nearest Neighbor

Classes in the vector space

Should the document ★ be assigned to China, UK or Kenya?

Classify the test document as the class of the document “nearest” to the query document (use vector similarity to find most similar doc)
Classification by kNN

Classify the test document as the majority class of the k documents “nearest” to the query document.

Classes in the vector space

China

UK

Kenya

Should the document ★ be assigned to China, UK or Kenya?
kNN classification

- kNN = $k$ nearest neighbors
kNN classification

- kNN = $k$ nearest neighbors
- For $k = 1$ (1NN), we assign each test document to the class of its nearest neighbor in the training set.
kNN classification

- $kNN = k$ nearest neighbors
- For $k = 1$ (1NN), we assign each test document to the class of its nearest neighbor in the training set.
- 1NN is not very robust – one document can be mislabeled or atypical.
kNN classification

- kNN = $k$ nearest neighbors
- For $k = 1$ (1NN), we assign each test document to the class of its nearest neighbor in the training set.
- 1NN is not very robust – one document can be mislabeled or atypical.
- For $k > 1$, we assign each test document to the majority class of its $k$ nearest neighbors in the training set.
kNN classification

- kNN = \( k \) nearest neighbors
- For \( k = 1 \) (1NN), we assign each test document to the class of its nearest neighbor in the training set.
- 1NN is not very robust – one document can be mislabeled or atypical.
- For \( k > 1 \), we assign each test document to the majority class of its \( k \) nearest neighbors in the training set.

Rationale of kNN: contiguity hypothesis
- We expect a test document \( d \) to have the same label as the training documents located in the local region surrounding \( d \).
Important Concepts

- **Data**: labeled instances, e.g. emails marked spam/ham
  - Training set
  - Held out set
  - Test set

- **Features**: attribute-value pairs which characterize each x

- **Experimentation cycle**
  - Learn parameters (e.g. model probabilities) on training set
  - (Tune hyperparameters on held-out set)
  - Compute accuracy of test set
  - Very important: never “peek” at the test set!

- **Evaluation**
  - Accuracy: fraction of instances predicted correctly

- **Overfitting and generalization**
  - Want a classifier which does well on test data
  - Overfitting: fitting the training data very closely, but not generalizing well
Classification by kNN

Classes in the vector space

China

UK

Kenya

Should the document ⭐ be assigned to China, UK or Kenya?

What are the features? What’s the training data? Testing data? Parameters?