Quiz #2

Topics

- Linguistics 101
- Parsing
  - Grammars, CFGs, PCFGs
  - Top-down vs. bottom-up
  - CKY algorithm
  - Grammar learning
  - Evaluation
  - Improved models
- Text similarity
- Will also be covered on Quiz #3, though

Admin

Assignment 4

Quiz #2 Thursday

- 1 hour (shouldn’t need that long)
- Will post link on piazza
- Will be available 12:15-1:15pm on class zoom
- Open book and notes
- Class starts at 1:15pm

Assignment 5 out soon

Text Similarity

A common question in NLP is how similar are texts

\[ \text{score: } \text{sim}(\text{ }, \text{ }) = ? \]

\[ \text{rank: } ? \]
Bag of words representation

For now, let’s ignore word order:

Obama said banana repeatedly last week on tv, “banana, banana, banana”

(4, 1, 1, 0, 0, 1, 0, 0, …)

“Bag of words representation”: multi-dimensional vector, one dimension per word in our vocabulary

Frequency of word occurrence

Vector based word

A
a: When 1
a: the 2
a: defendant 1
a: and 1
a: courthouse 0
...

Multidimensional vectors, one dimension per word in our vocabulary

B
b: When 1
b: the 2
b: defendant 1
b: and 0
b: courthouse 1
...

Normalized distance measures

Cosine

\[ \text{sim}_{\cos}(A, B) = \frac{A \cdot B}{\sqrt{\sum a_i^2 \cdot \sum b_i^2}} \]

L2

\[ \text{dist}_{L2}(A, B) = \sqrt{\sum (a_i' - b_i')^2} \]

L1

\[ \text{dist}_{L1}(A, B) = \sum |a_i' - b_i'| \]

\(a'\) and \(b'\) are length normalized versions of the vectors

Our problems

Which of these have we addressed?

- word order
- length
- synonym
- spelling mistakes
- word importance
- word frequency
Word overlap problems

Treats all words the same

A: When the defendant and his lawyer walked into the court, some of the victim supporters turned their backs to him.

B: When the defendant walked into the courthouse with his attorney, the crowd turned their backs on him.

Ideas?

Word importance

Include a weight for each word/feature

A

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>ai</td>
<td>1</td>
<td>w1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>as</td>
<td>2</td>
<td>w2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>au</td>
<td>1</td>
<td>w3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>and</td>
<td>1</td>
<td>w4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>0</td>
<td>w5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

B

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>bi</td>
<td>1</td>
<td>w1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bs</td>
<td>2</td>
<td>w2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bs</td>
<td>1</td>
<td>w3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>and</td>
<td>0</td>
<td>w4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bs</td>
<td>1</td>
<td>w5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Distance + weights

We can incorporate the weights into the distances

Think of it as either (both work out the same):

- preprocessing the vectors by multiplying each dimension by the weight
- incorporating it directly into the similarity measure

\[
sim_{cos}(A, B) = \frac{\sum_i w_i a_i w_i b_i}{\sqrt{\sum_i (w_i a_i)^2} \sqrt{\sum_i (w_i b_i)^2}}
\]

Idea: use corpus statistics

the
defendant

What would be a quantitative measure of word importance?
Document frequency

**Document frequency (DF)** is one measure of word importance

Terms that occur in many documents are weighted less, since overlapping with these terms is very likely

- In the extreme case, take a word like *the* that occurs in almost EVERY document

Terms that occur in only a few documents are weighted more

---

Document vs. overall frequency

The overall frequency of a word is the number of occurrences in a dataset, counting multiple occurrences

**Example:**

<table>
<thead>
<tr>
<th>Word</th>
<th>Overall frequency</th>
<th>Document frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>insurance</td>
<td>10440</td>
<td>3997</td>
</tr>
<tr>
<td>try</td>
<td>10422</td>
<td>8760</td>
</tr>
</tbody>
</table>

Which word is a more informative (and should get a higher weight)?

---

From document frequency to weight

weight and document frequency are inversely related
- Higher document frequency should have lower weight and vice versa

Document frequency is unbounded
- Document frequency will change depending on the size of the data set (i.e. the number of documents)

---

\[
sim_{cos}(A, B) = \frac{A \cdot B}{\sqrt{\sum_x (A_x)^2} \sqrt{\sum_y (B_y)^2}}
\]
Inverse document frequency

\[ \text{idf}_w = \log \frac{N}{df_w} \]

IDF is inversely correlated with DF
- higher DF results in lower IDF

N incorporates a dataset dependent normalizer
- log dampens the overall weight

IDF example, suppose \( N = 1 \text{ million} \)

<table>
<thead>
<tr>
<th>term</th>
<th>df</th>
<th>idf</th>
</tr>
</thead>
<tbody>
<tr>
<td>calpurnia</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>animal</td>
<td>100</td>
<td>4</td>
</tr>
<tr>
<td>sunday</td>
<td>1,000</td>
<td>3</td>
</tr>
<tr>
<td>fly</td>
<td>10,000</td>
<td>2</td>
</tr>
<tr>
<td>under</td>
<td>100,000</td>
<td>1</td>
</tr>
<tr>
<td>the</td>
<td>1,000,000</td>
<td>0</td>
</tr>
</tbody>
</table>

What are the IDFs assuming log base 10?

IDF example, suppose \( N = 1 \text{ million} \)

<table>
<thead>
<tr>
<th>term</th>
<th>df</th>
<th>idf</th>
</tr>
</thead>
<tbody>
<tr>
<td>calpurnia</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>animal</td>
<td>100</td>
<td>4</td>
</tr>
<tr>
<td>sunday</td>
<td>1,000</td>
<td>3</td>
</tr>
<tr>
<td>fly</td>
<td>10,000</td>
<td>2</td>
</tr>
<tr>
<td>under</td>
<td>100,000</td>
<td>1</td>
</tr>
<tr>
<td>the</td>
<td>1,000,000</td>
<td>0</td>
</tr>
</tbody>
</table>

What if we didn't use the log to dampen the weighting?

There is one idf value/weight for each word
IDF example, suppose $N=1$ million

<table>
<thead>
<tr>
<th>term</th>
<th>df</th>
<th>idf</th>
</tr>
</thead>
<tbody>
<tr>
<td>calpurnia</td>
<td>1</td>
<td>1,000,000</td>
</tr>
<tr>
<td>animal</td>
<td>100</td>
<td>10,000</td>
</tr>
<tr>
<td>sunday</td>
<td>1,000</td>
<td>1,000</td>
</tr>
<tr>
<td>fly</td>
<td>10,000</td>
<td>100</td>
</tr>
<tr>
<td>under</td>
<td>100,000</td>
<td>10</td>
</tr>
<tr>
<td>the</td>
<td>1,000,000</td>
<td>1</td>
</tr>
</tbody>
</table>

What if we didn’t use the log to dampen the weighting?

TF-IDF

One of the most common weighting schemes

$TF = \text{term frequency}$

$IDF = \text{inverse document frequency}$

$a_i' = a_i \times \log \frac{N}{df_i}$

We can then use this with any of our similarity measures!

Stoplists: extreme weighting

Some words like ‘a’ and ‘the’ will occur in almost every document

- IDF will be 0 for any word that occurs in all documents
- For words that occur in almost all of the documents, they will be nearly 0

A stoplist is a list of words that should not be considered (in this case, similarity calculations)

- Sometimes this is the n most frequent words
- Often, it’s a list of a few hundred words manually created

Stoplist

- If most of these end up with low weights anyway, why use a stoplist?
Stoplists

Two main benefits
- More fine grained control: some words may not be frequent, but may not have any content value (alas, teh, gosh)
- Often does contain many frequent words, which can drastically reduce our storage and computation

Any downsides to using a stoplist?
- For some applications, some stop words may be important

Our problems

Which of these have we addressed?
- word order
- length
- synonym
- spelling mistakes
- word importance
- word frequency

A model of word similarity!

Word overlap problems

A: When the defendant and his lawyer walked into the court, some of the victim supporters turned their backs to him.

B: When the defendant walked into the courthouse with his attorney, the crowd turned their backs on him.

Word similarity

How similar are two words?

score: \( \text{sim}(w_1, w_2) = ? \)

rank: \( w_? \)

list: \( w_1 \) and \( w_2 \) are synonyms
Word similarity applications

- General text similarity
- Thesaurus generation
- Automatic evaluation
- Text-to-text
  - paraphrasing
  - summarization
  - machine translation
- Information retrieval (search)

Word similarity

How similar are two words?

score: \( \text{sim}(w_1, w_2) = ? \)

rank: \( w_1 \) \( w_2 \) \( w_3 \)

list: \( w_1 \) and \( w_2 \) are synonyms

Character-based similarity

\( \text{sim}(\text{turned}, \text{truned}) = ? \)

How might we do this using only the words (i.e. no outside resources?)
Edit distance (Levenshtein distance)

The edit distance between \( w_1 \) and \( w_2 \) is the minimum number of operations to transform \( w_1 \) into \( w_2 \).

Operations:
- insertion
- deletion
- substitution

\[
\begin{align*}
\text{EDIT}(\text{turned, truned}) &= \text{?} \\
\text{EDIT}(\text{computer, commuter}) &= \text{?} \\
\text{EDIT}(\text{banana, apple}) &= \text{?} \\
\text{EDIT}(\text{wombat, worcester}) &= \text{?}
\end{align*}
\]

Better edit distance

Are all operations equally likely?
- No

Improvement: give different weights to different operations
- replacing a for e is more likely than z for y

Ideas for weightings?
- Learn from actual data (known typos, known similar words)
- Intuitions: phonetics
- Intuitions: keyboard configuration

Vector character-based word similarity

\[
sim(\text{turned, truned}) = \text{?}
\]

Any way to leverage our vector-based similarity approaches from last time?
Vector character-based word similarity

\[ \text{sim}(\text{turned, truned}) = ? \]

- Character level loses a lot of information

Similiarity problems?

Vector character-based word similarity

\[ \text{sim}(\text{restful, fluster}) = ? \]

- Use character bigrams or even trigrams

Word similarity

Four general categories

- Character-based
  - turned vs. truned
  - cognates (night, nacht, nicht, natt, nat, no, noch)
- Semantic web-based (e.g. WordNet)
- Dictionary-based
- Distributional similarity-based
  - similar words occur in similar contexts
Word similarity

Four general categories
- Character-based
  - turned vs. truned
  - cognates (night, nicht, nicht, att, atc, noc, noch)
- Semantic web-based (e.g. WordNet)
- Dictionary-based
- Distributional similarity-based
  - similar words occur in similar contexts

Dictionary-based similarity

Utilize our text similarity measures

\[
\text{sim}(\text{dog}, \text{beagle}) = \text{sim}(\text{dog}, \text{beagle})
\]

One of a breed of small hounds having long ears, short legs, and a usually black, tan, and white coat.

Any carnivore of the family Canidae, having prominent canine teeth and, in the wild state, a long and slender muzzle, a deep-chested muscular body, a bushy tail, and large, erect ears. Compare canid.

Dictionary-based similarity

Word
- aardvark
  - a large, nocturnal, burrowing mammal, Orycteropus afer, of central and southern Africa, feeding on ants and termites and having a long, extensile tongue, strong claws, and long ears.
- beagle
  - One of a breed of small hounds having long ears, short legs, and a usually black, tan, and white coat.
- dog
  - Any carnivore of the family Canidae, having prominent canine teeth and, in the wild state, a long and slender muzzle, a deep-chested muscular body, a bushy tail, and large, erect ears. Compare canid.

What about words that have multiple senses/parts of speech?
Dictionary-based similarity

1. part of speech tagging
2. word sense disambiguation
3. most frequent sense
4. average similarity between all senses
5. max similarity between all senses
6. sum of similarity between all senses

Dictionary + WordNet

WordNet also includes a “gloss” similar to a dictionary definition

Other variants include the overlap of the word senses as well as those word senses that are related (e.g. hypernym, hyponym, etc.)

- incorporates some of the path information as well
- Banerjee and Pedersen, 2003

Word similarity

Four general categories

- Character-based
  - turned vs. truned
  - cognates (night, nacht, natt, nat, noc, noch)
- Semantic web-based (e.g. WordNet)
- Dictionary-based
  - similar words occur in similar contexts

Corpus-based approaches

Word

<table>
<thead>
<tr>
<th>Word</th>
<th>ANY blurb with the word</th>
</tr>
</thead>
<tbody>
<tr>
<td>aardvark</td>
<td></td>
</tr>
<tr>
<td>beagle</td>
<td></td>
</tr>
<tr>
<td>dog</td>
<td>Ideas?</td>
</tr>
</tbody>
</table>
The Beagle is a breed of small to medium-sized dog. A member of the Hound Group, it is similar in appearance to the Foxhound but smaller, with shorter legs. Beagles are intelligent, and are popular as pets because of their size, even temper, and lack of inherited health problems.

Dogs of similar size and purpose to the modern Beagle can be traced in Ancient Greece[2] back to around the 5th century BC. From medieval times, beagle was used as a generic description for the smaller hounds, though these dogs differed considerably from the modern breed.

In the 1840s, a standard Beagle type was beginning to develop: the distinction between the North Country Beagle and Southern...
Corpus-based similarity

$$\text{sim}(\text{dog, beagle}) =$$

$$\text{sim}(\text{context\_vector(dog), context\_vector(beagle)})$$

the: 5 the: 2
is: 1 is: 1
as: 4 as: 2
breeds: 2 breed: 1
are: 1 are: 1
intelligent: 5 intelligent: 1
... to: 1 modern: 1
...

Web-based similarity

Ideas?

Concatenate the snippets for the top $N$ results

Concatenate the web page text for the top $N$ results
Another feature weighting

TF-IDF weighting takes into account the general importance of a feature.

For distributional similarity, we have the feature \( f \), but we also have the word itself \( w \) that we can use for information.

\[
\text{sim}(\text{context}_\text{vector}(\text{dog}), \text{context}_\text{vector}(\text{beagle}))
\]

\[
\begin{align*}
\text{the} & : 5 & \text{the} & : 2 \\
\text{is} & : 1 & \text{is} & : 1 \\
\text{breeds} & : 2 & \text{breeds} & : 1 \\
\text{are} & : 1 & \text{are} & : 1 \\
\text{intelligent} & : 5 & \text{intelligent} & : 1 \\
\text{and} & : 1 & \text{and} & : 1 \\
\text{to} & : 1 & \text{to} & : 1 \\
\text{modern} & : 1 & \text{modern} & : 1 \\
\end{align*}
\]

Feature weighting ideas given this additional information?

\[
\text{sim}(\text{context}_\text{vector}(\text{dog}), \text{context}_\text{vector}(\text{beagle}))
\]

\[
\begin{align*}
\text{the} & : 3 & \text{the} & : 2 \\
\text{is} & : 1 & \text{is} & : 1 \\
\text{breeds} & : 2 & \text{breeds} & : 1 \\
\text{are} & : 1 & \text{are} & : 1 \\
\text{intelligent} & : 5 & \text{intelligent} & : 1 \\
\text{and} & : 1 & \text{and} & : 1 \\
\text{to} & : 1 & \text{to} & : 1 \\
\text{modern} & : 1 & \text{modern} & : 1 \\
\end{align*}
\]

Mutual information

A bit more probability 😊

\[
I(X,Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}
\]

When will this be high and when will this be low?
Mutual information

A bit more probability 😊

\[ I(X,Y) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \]

if \( x \) and \( y \) are independent (i.e. one occurring doesn't impact the other occurring) then:

\[ p(x,y) = p(x)p(y) \]

61

Mutual information

A bit more probability 😊

\[ I(X,Y) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \]

if \( x \) and \( y \) are independent (i.e. one occurring doesn't impact the other occurring) then:

\[ p(x,y) = p(x)p(y) \]

What does this do to the sum?

62

Mutual information

A bit more probability 😊

\[ I(X,Y) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \]

if they are dependent then:

\[ p(x,y) = p(x)p(y|x) = p(y)p(x|y) \]

\[ I(X,Y) = \sum_x \sum_y p(x,y) \log \frac{p(y|x)}{p(y)} \]

63

Mutual information

A bit more probability 😊

\[ I(X,Y) = \sum_x \sum_y p(x,y) \log \frac{p(y|x)}{p(y)} \]

What is this asking?
When is this high?

How much more likely are we to see \( y \) given \( x \) has a particular value!

64
### Point-wise mutual information

**Mutual information**

$$I(X,Y) = \sum_x \sum_y p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

How related are two variables (i.e. over all possible values/events)

**Point-wise mutual information**

$$PMI(x,y) = \log \frac{p(x,y)}{p(x)p(y)}$$

How related are two particular events/values

---

### PMI weighting

Mutual information is often used for feature selection in many problem areas

PMI weighting weights co-occurrences based on their correlation (i.e. high PMI)

**context_vector (beagle)**

<table>
<thead>
<tr>
<th></th>
<th>PMI(x,y)</th>
</tr>
</thead>
<tbody>
<tr>
<td>the</td>
<td>2</td>
</tr>
<tr>
<td>its</td>
<td>1</td>
</tr>
<tr>
<td>is</td>
<td>2</td>
</tr>
<tr>
<td>breed</td>
<td>1</td>
</tr>
<tr>
<td>are</td>
<td>1</td>
</tr>
<tr>
<td>intelligent</td>
<td>1</td>
</tr>
<tr>
<td>and</td>
<td>1</td>
</tr>
<tr>
<td>to</td>
<td>1</td>
</tr>
<tr>
<td>modern</td>
<td>1</td>
</tr>
</tbody>
</table>

How do we calculate these?