Assignment 2
CMSC 473/673 — Introduction to Natural Language Processing

Due Wednesday September 18th, 2019, 11:59 PM

<table>
<thead>
<tr>
<th>Item</th>
<th>Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assigned</td>
<td>Wednesday September 4th, 2019</td>
</tr>
<tr>
<td>Due</td>
<td>Wednesday September 18th, 2019</td>
</tr>
<tr>
<td>Topic</td>
<td>Probabilities and Language Modeling</td>
</tr>
<tr>
<td>Points</td>
<td>105</td>
</tr>
</tbody>
</table>

In this assignment you will gain additional experience with probabilities, implement more counting techniques, and use them to explore a corpus.

You are to complete this assignment on your own: that is, the code and writeup you submit must be entirely your own. However, you may discuss the assignment at a high level with other students or on the discussion board. Note at the top of your assignment who you discussed this with or what resources you used (beyond course staff, any course materials, or public Piazza discussions).

The following table gives the overall point breakdown for this assignment.

<table>
<thead>
<tr>
<th>Question</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Points</td>
<td>10</td>
<td>20</td>
<td>30</td>
<td>15</td>
<td>30</td>
</tr>
</tbody>
</table>

You have incredible freedom in how you structure your code. Your code does not have to be pristine. However, we should be be able to follow what you write. Ensure your code is appropriately commented and described in your writeup. Try to make the coding easy on yourself.

What To Turn In  Turn in a writeup in PDF format that answers the questions; turn in all requested code necessary to replicate your results. Be sure to include specific instructions on how to build (compile) your code. Answers to the following questions should be long-form. Provide any necessary analyses and discussion of your results.

How To Submit  Submit the assignment on the submission site:

[https://www.csee.umbc.edu/courses/undergraduate/473/f19/submit](https://www.csee.umbc.edu/courses/undergraduate/473/f19/submit)

Be sure to select “Assignment 2.”
Full Questions

1. (10 points) Let $X$ and $Y$ be random variables where there’s at least one pair of values $x$ and $y$ such that $P(X = x, Y = y) = 0$. Show, with justification, whether $X$ and $Y$ are independent. (The exact values $x$ and $y$ do not matter.)

2. (20 points) Let’s return to the dice rolling problem from A1Q2: let $X_i$ be the random variable representing the $i$th role of a six-sided die and let $x_i$ be the rolled (observed) value of the $i$th roll. Assume that we roll the die $N$ times and that each $X_i$ is an i.i.d. sample.

(A) Because each $X_i$ is i.i.d., we can work with a single distribution $p(X = k)$. Define $c_k$ as the number of times that we observed a $k$ among our $N$ rolls:

$$c_k = \sum_{i=1}^{N} \left(1 \text{ if } X_i = k \text{ otherwise } 0\right).$$

Define $p(X = k) = \frac{c_k}{Z}$, where $Z$ is a constant that ensures $p$ sums to one:

$$\sum_k p(X = k) = \sum_k \frac{c_k}{Z} = 1.$$ (2)

Compute $Z$.

(B) Let’s say that before we rolled our die, we first flipped a fair coin (2 outcomes, where probability of heads equals probability of tails). Call this random variable $Z_i$. If $Z_i = H$ (heads), then we roll a fairly weighted, six-sided die. If $Z_i = T$ (tails), then we roll a six-sided die where the probabilities of rolling each outcome is given by the probability vector $\pi = \left(\frac{1}{9}, \frac{2}{9}, \frac{1}{9}, \frac{2}{9}, \frac{1}{9}, \frac{2}{9}\right)$.

i. Write the conditional distributions $p(X | Z)$. You can think of this as filling out the table

| $p(X | Z)$ | $X = 1$ | $X = 2$ | $X = 3$ | $X = 4$ | $X = 5$ | $X = 6$ |
|------------|---------|---------|---------|---------|---------|---------|
| $Z = H$    |         |         |         |         |         |         |
| $Z = T$    |         |         |         |         |         |         |

(Hint: You should write out two different distributions.)

ii. Now write the joint distribution $p(X, Z)$. You can think of this as filling out the table

<table>
<thead>
<tr>
<th>$p(X, Z)$</th>
<th>$X = 1$</th>
<th>$X = 2$</th>
<th>$X = 3$</th>
<th>$X = 4$</th>
<th>$X = 5$</th>
<th>$X = 6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Z = H$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$Z = T$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Hint: Think about the number of different distributions you need to write out.)

iii. Does $p(X, Z) = p(Z, X)$?

iv. One way we can define the marginal distribution of $x_i$ is as

$$p(X_i = k) = \sum_j p(X_i = k | Z_i = j)p(Z_i = j).$$ (3)

(Though there may be equivalent ways of computing it!) I tell you that after four rolls the die came up as 3, 1, 6, and 5. How likely is that? (That is, what are the marginal probability values $p(X_1 = 3)$, $p(X_2 = 1)$, $p(X_3 = 6)$, and $p(X_4 = 5)$?)

v. Show, using the definition of probabilistic independence, whether $X$ and $Z$ are independent.

3. (30 points) Read and write a half page summary and review of Church and Hanks (1989); see the below citation for a URL. Your summary should discuss the basic methodology and findings of this

---

1 Single spaced, regular font, and one column is fine.
paper. In addition, identify findings you found interesting, surprising, or confusing. Try to answer the question, “What is the overall takeaway (for you) from this paper?”

The full Bibtex citation is

@InProceedings{church-hanks:1989:ACL,
  author = {Church, Kenneth Ward and Hanks, Patrick},
  title = {Word Association Norms, Mutual Information, andLexicography},
  booktitle = {Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics},
  month = {June},
  year = {1989},
  address = {Vancouver, British Columbia, Canada},
  publisher = {Association for Computational Linguistics},
  pages = {76--83},
  url = {http://www.aclweb.org/anthology/P89-1010},
  doi = {10.3115/981623.981633}
}

4. (15 points) It’s common to say that language follows a power law distribution. Alternatively, you may hear that ‘language is Zipfian.’ In this question, you’ll explore what that means, and examine just how Zipfian two different languages can be.

First, imagine counting up words and performing a descending sort of the words according to how many times each appeared. That is, words that appear more often appear before words that appear less often—in such a setting, the more common words are said to have a lower rank. The basic Zipf estimate states that the frequency $f(y)$ of a word $y$ is inversely proportional to how common it is (its rank $r$). Taking logarithms turns this into a linear relationship (in log-log space):

$$\log f(y) = C - m \log r(y), \quad (4)$$

where $C$ and $m$ are constants. You can define $f(y)$ as either the count of $y$, or the normalized count, as in (2)—whichever’s easiest for you.

Explore how well Zipf’s law holds up on two languages, A and B, from the Universal Dependency data.

- First, pick your languages A and B.
- Second, compute $f_X(y)$ and $r_X(y)$ for each word type $y$ and language $X$.
- Third, plot $\log f_X(y)$ vs. $\log r_X(y)$ as a scatterplot.
- Fourth, perform a linear regression (ordinary least squares), which fits a line to provided independent and dependent variables.
- Fifth, include these plots in your writeup and provide a discussion of what you found. What does the slope of the linear regression tell you? What about the coefficient of determination (the $R^2$ value)? In your analysis, include a description of your methodology, the results, and an analysis of the results.

Remember to turn in all code/artifacts needed to reproduce your analysis. Though we’ll have your code, your methodology description should be thorough enough for us to reimplement what you did.

---

2 To perform this linear regression, you may use any external library, including Excel. In R, linear regression is available through the \texttt{lm} function, from the basic \texttt{stats} package (typically preloaded). Linear regression is available in Python through a number of methods, including from \texttt{scipy} under \texttt{scipy.stats.linregress}. 
5. **(30 points)** In this question, you are going to implement the PMI (pointwise mutual information) measure \( I(x, y) \) from Church and Hanks (1989) and explore it on the UD\_English-EWT training portion. Define each sentence as the “window.” (If two words \( x \) and \( y \) appear in the same sentence, they impact \( p(x, y) \). Otherwise they don’t.)

Let \( \nu \) and \( \eta \) be any two arbitrary word types. To implement PMI, we’ll refer to \( p(\nu), p(\eta), \) and \( p(\nu, \eta) \). Church and Hanks discuss two options for \( p(\nu, \eta) \). The first is a symmetric measure that measures how likely \( \nu \) and \( \eta \) appear in the same “window,” without caring about the order. The second is an asymmetric measure that measures how likely \( \eta \) follows \( \nu \) (within a “window”). For this question, you may answer using either of them. (It is your choice.)

(A) Are all of those “\( p \)” functions the same? If so, explain why. If not, identify which ones are the same and which are different.

(B) What values do you use to normalize \( p(\nu), p(\eta), \) and \( p(\nu, \eta) \)?

(C) Implement these functions \( p(\nu), p(\eta), \) and \( p(\nu, \eta) \).

   i. What are the ten words \( \nu \) that have the ten highest values \( p(\nu) \)?

   ii. What are the ten words \( \eta \) that have the ten highest values \( p(\eta) \)?

   iii. What are the ten pairs of words \( \nu, \eta \) that have the ten highest values \( p(\nu, \eta) \)?

   (If any of the \( p \) distributions are identical, incorporate this into your answer.)

(D) Use those functions to implement PMI.

   i. Do you agree with their rough rule-of-thumb that “interesting” word pairs \( (x, y) \) have \( I(x, y) > 3 \)? Provide some examples to help support your assessment.

   ii. What are the top 10 \( I(\nu, \eta) \) associations with the word “doctor?”

   iii. What are the bottom 10 \( I(\nu, \eta) \) associations with the word “doctor?”

(E) Church and Hanks define \( p(\nu), p(\eta), p(\nu, \eta) \) in terms of their association counts. We’ll call those \( g(\nu), g(\eta), g(\nu, \eta) \). Now let \( \lambda \) be any positive (real) number: it could be 0.000001, 10, 203983204981, or any other valid number. Define \( g_\lambda = g + \lambda \), and redefine \( p \) in terms of \( g_\lambda \). Call this new function \( p_\lambda \).

   i. If \( g(\text{cat}) = 42 \) (which it does), and there are 204607 tokens, what is \( p_\lambda(\text{cat}) \) if \( \lambda = 3 \).

   ii. Why can’t \( \lambda \) be negative?

   iii. Now re-implement \( I \) in terms of \( p_\lambda \) (letting \( \lambda \) be a variable, i.e., it is not always 3). Try 5 values of \( \lambda \): two values below 1 (\( \lambda_1, \lambda_2 < 1 \)), one value at 1 (\( \lambda_3 = 1 \)), and two values above 1 (\( \lambda_4, \lambda_5 > 1 \)). How much do the top 10 \( I(\nu, \eta) \) words change as you decrease, or increase, \( \lambda \).

(F) Using the above results (and any other results you feel you need to gather to answer this), discuss potential implications for the “power” (ability to find “interesting word pairs”) and “stability” (consistency/variability in results) of these methods. What are some of the potential risks in using a method like this to find interesting words, and how could you mitigate that risk?