Natural Language Processing
Assignment 2: Probability and Vector Exercises

Li-Yi Lin, JHED ID: 5F0CE8

Question 1.
(a) Answer:
Since $Y \subseteq Z$, we have $Z = Y \cup (Z \cap \neg Y)$. So the probability of Z is

$$\begin{aligned} p(Z) &= p(Y \cup (Z \cap \neg Y)) \\ &= p(Y) + p(\emptyset) \\ &\geq p(Y) + 0 && (p(\emptyset) \geq 0) \\ &= p(Y) \end{aligned}$$

Thus, we proved that if $Y \subseteq Z$, then $p(Y) \leq p(Z)$.

(b) Answer:
We know that $p(X \mid Z) = \frac{p(X \cap Z)}{p(Z)}$. Since $(X \cap Z) \subseteq Z$, we have the fact that $p(X \cap Z) \leq p(Z)$. Therefore, $p(X \mid Z) = \frac{p(X \cap Z)}{p(Z)} \leq 1$. Moreover, since probability is larger than or equal to 0, we have $p(X \cap Z) \geq 0$. Because $p(X \mid Z)$ is conditioned on Z, we have $p(Z) > 0$. Therefore, by the property that $0 \leq p(X \cap Z) \leq p(Z)$, we can prove that $p(X \mid Z)$ always fall in the range [0,1].

(c) Answer:
Since $E \cap \emptyset = \emptyset$, we have $p(E \cup \emptyset) = p(E) + p(\emptyset)$ by the given axioms. Because we know that $p(E) = 1$ and probability cannot be larger than 1, $p(E \cup \emptyset) = p(E) + p(\emptyset) = 1 + p(\emptyset) \leq 1$. Therefore $p(\emptyset)$ must be 0.

(d) Answer:
Let $\bar{X}$ denote $E - X$. We have $\bar{X} \cup X = E$. Now we are going to prove that $p(X) = 1 - p(\bar{X})$.
Since $\bar{X} \cap X = \emptyset$ and $\bar{X} \cup X = E$, we have

$$\begin{aligned} p(E) &= p(\bar{X} \cup X) \\ &= p(\bar{X}) + p(X) - p(\bar{X} \cap X) \\ &= p(\bar{X}) + p(X) && (p(\bar{X} \cap X) = 0 \text{ by (c)}) \end{aligned}$$

By the equation above, we have $p(E) = p(\bar{X}) + p(X)$. Since $p(E) = 1$, we have $1 = p(\bar{X}) + p(X)$. Therefore we have proven that $p(X) = 1 - P(\bar{X})$.

(e) Answer:

$$\begin{aligned} p(singing \cap rainy \mid rainy) &= \frac{p((singing \cap rainy) \cap rainy)}{p(rainy)} \\ &= \frac{p(singing \cap (rainy \cap rainy))}{p(rainy)} && \text{(Intersection is an associative operation)} \\ &= \frac{p(singing \cap rainy)}{p(rainy)} && (rainy \cap rainy = rainy) \\ &= p(singing \mid rainy) \end{aligned}$$

By the equation above, we proved that $p(singing \text{ AND } rainy \mid rainy) = p(singing \mid rainy)$.

(f) Answer:

$$\begin{aligned} p(Y) &= p((X \cap Y) \cup (\bar{X} \cap Y)) \\ &= p(X \cap Y) + p(\bar{X} \cap Y) - p((X \cap Y) \cap (\bar{X} \cap Y)) \\ &= p(X \cap Y) + p(\bar{X} \cap Y) && ((X \cap Y) \cap (\bar{X} \cap Y) = \emptyset) \end{aligned}$$

By dividing both sides by $p(Y)$, we proved that

$$1 = \frac{p(X \cap Y)}{p(Y)} + \frac{p(\bar{X} \cap Y)}{p(Y)}$$
$$p(X \mid Y) = 1 - p(\bar{X} \mid Y)$$

(g) Answer:

$$\begin{aligned} & (p(X \mid Y) \cdot p(Y) + p(X \mid \bar{Y}) \cdot p(\bar{Y})) \cdot p(\bar{Z} \mid X)/p(\bar{Z}) \\ = & (\frac{p(X \cap Y)}{p(Y)} \cdot p(Y) + \frac{p(X \cap \bar{Y})}{p(\bar{Y})} \cdot p(\bar{Y})) \cdot \frac{p(\bar{Z} \cap X)}{p(X)}/p(\bar{Z}) \\ = & (p(X \cap Y) + p(X \cap \bar{Y})) \cdot \frac{p(\bar{Z} \cap X)}{p(X)}/p(\bar{Z}) \\ = & p(X) \cdot \frac{p(\bar{Z} \cap X)}{p(X)}/p(\bar{Z}) \\ = & \frac{p(\bar{Z} \cap X)}{p(\bar{Z})} \\ = & p(X \mid \bar{Z}) \end{aligned}$$

(h) Answer:
We know that

$$p(singing \text{ AND } rainy) = p(singing) + p(rainy) - p(singing \cap rainy)$$

So, when $(singing \cap rainy) = \emptyset$, then

$$\begin{aligned} p(singing \text{ AND } rainy) &= p(singing) + p(rainy) - p(singing \cap rainy) \\ &= p(singing) + p(rainy) - p(\emptyset) \\ &= p(singing) + p(rainy) - 0 && \text{(by (c))} \\ &= p(singing) + p(rainy) \end{aligned}$$

Therefore, when set *singing* and *rainy* are disjoint, the given equation is true.

(i) Answer:
Assume the given equation is true, then

$$p(singing \text{ AND } rainy) = p(singing) \cdot p(rainy)$$

We further assume that $p(rainy) > 0$ and divide the above equation by $p(rainy)$

$$\frac{p(singing \text{ AND } rainy)}{p(rainy)} = p(singing)$$

$$p(singing \mid rainy) = p(singing)$$

Therefore, for the $p(singing \text{ AND } rainy) = p(singing) \cdot p(rainy)$ to be true, $p(singing \mid rainy) = p(singing)$ must also be true. It means that the probability event *rainy* will not affect the probability of event *singing*.

(j) Answer:
We are given that

$$p(X \mid Y) = 0$$
$$\frac{p(X \cap Y)}{p(Y)} = 0$$

For the above equation to be 0, $p(X \cap Y)$ must be 0.

Then, we are going to prove that $p(X \mid Y, Z) = 0$

$$\begin{aligned} p(X \mid Y, Z) &= \frac{p(X \cap Y \cap Z)}{p(Y \cap Z)} \\ &\leq \frac{p(X \cap Y)}{p(Y \cap Z)} && \text{(by (a), } (X \cap Y \cap Z) \subseteq (X \cap Y) \text{, so } p(X \cap Y \cap Z) \leq p(X \cap Y)) \\ &= \frac{0}{p(Y \cap Z)} \\ &= 0 \end{aligned}$$

Thus, we have proved that if $p(X \mid Y) = 0$, then $p(X \mid Y, Z) = 0$.

(k) Answer:
By (f), we know

$$\begin{aligned} p(\bar{W} \mid Y) &= 1 - p(W \mid Y) \\ &= 1 - 1 \\ &= 0 \end{aligned}$$

Since $p(\bar{W} \mid Y) = \frac{p(\bar{W} \cap Y)}{p(Y)} = 0$, then $p(\bar{W} \mid Y)$ must be 0.

Then, we are going to prove that $p(W \mid Y, Z) = 1$

$$\begin{aligned} p(W \mid Y, Z) &= 1 - p(\bar{W} \mid Y, Z) && \text{(by (f))} \\ &= 1 - \frac{p(\bar{W} \cap Y \cap Z)}{p(Y \cap Z)} \\ &= 1 - 0 && \text{(since } (\bar{W} \cap Y \cap Z) \subseteq (\bar{W} \cap Y), p(\bar{W} \cap Y \cap Z) \leq p(\bar{W} \cap Y) \text{ by (a))} \\ &= 1 \end{aligned}$$

Thus, we have proven that if $p(W \mid Y) = 1$, then $p(W \mid Y, Z) = 1$.

Question 2. (a) Answer:
Denote event "$Actual = Blue$" as A, "$Claimed = blue$" as C.
By Bayes' theorem, we have

$$p(A \mid C)p(C) = p(C \mid A)p(A)$$

(b) Answer:
Since $p(Actual = blue) = 0.1$ is fixed. We further rewrite the equation in (a) as shown below:

$$p(Actual = blue \mid Claimed = blue) \propto p(Claimed = blue \mid Actual = blue)p(Actual = blue)$$

Then, the prior probability is $p(Actual = blue)$,
the likelihood of the evidence is $p(Claimed = blue \mid Actual = blue)$,
and the posterior probability is $p(Actual = blue \mid Claimed = blue)$.

(c) Answer:
prior probability = $p(Actual = blue) = 0.1$
likelihood of the evidence = $p(Claimed = blue \mid Actual = blue) = 0.8$
posterior probability = $\frac{p(Claimed=blue|Actual=blue)\cdot p(Actual=blue)}{p(Claimed=blue)} = \frac{0.8\times 0.1}{p(Claimed=blue)}$

Now we need to calculate $p(Claimed = blue)$.
$p(Claimed = blue)$
$= p(Claimed = blue \mid Actual = blue)\cdot p(Actual = blue)+p(Claimed = blue \mid Actual = red)\cdot p(Actual = red)$
$= 0.8 \times 0.1 + 0.2 \times 0.9$
$= 0.26$

So, posterior probability = $\frac{0.8\times 0.1}{p(Claimed=blue)} = \frac{0.08}{0.26} = 0.3077$

The judge should care about the posterior because it takes not only about the observed events, likelihood, but also the belief about the probability of the event we want to guess. By considering the posterior probability, the judge can avoid the situation that our belief is wrong or we have not enough observed events so that the judge can have more accurate information for judgement.

(d) Answer:
To prove the given equation, we start with the right hand side:

$$\begin{aligned}\frac{p(B \mid A,Y)\cdot p(A \mid Y)}{p(B \mid Y)} &= \frac{\frac{p(B\cap A\cap Y)}{p(A\cap Y)}\cdot\frac{p(A\cap Y)}{p(Y)}}{\frac{p(B\cap Y)}{p(Y)}} \\ &= \frac{p(B\cap A\cap Y)}{p(B\cap Y)} \\ &= p(A \mid B,Y)\end{aligned}$$

Thus, by we have proven that

$$p(A \mid B,Y) = \frac{p(B \mid A,Y)\cdot p(A \mid Y)}{p(B \mid Y)}$$

(e) Answer:
By (d), we have

$$p(A \mid B,Y) = \frac{p(B \mid A,Y)\cdot p(A \mid Y)}{p(B \mid Y)}$$

Now we want to prove that

$$p(A \mid B, Y) = \frac{p(B \mid A, Y) \cdot p(A \mid Y)}{p(B \mid A, Y) \cdot p(A \mid Y) + p(B \mid \bar{A}, Y) \cdot p(\bar{A} \mid Y)}$$

So we only have to prove that

$$p(B \mid Y) = p(B \mid A, Y) \cdot p(A \mid Y) + p(B \mid \bar{A}, Y) \cdot p(\bar{A} \mid Y)$$

We start from the right hand side.

$$\begin{aligned} p(B \mid A, Y) \cdot p(A \mid Y) + p(B \mid \bar{A}, Y) \cdot p(\bar{A} \mid Y) &= \frac{p(B \cap A \cap Y)}{p(A \cap Y)} \cdot \frac{p(A \cap Y)}{p(Y)} + \frac{p(B \cap \bar{A} \cap Y)}{p(\bar{A} \cap Y)} \cdot \frac{p(\bar{A} \cap Y)}{p(Y)} \\ &= \frac{p(B \cap A \cap Y)}{p(Y)} + \frac{p(B \cap \bar{A} \cap Y)}{p(Y)} \\ &= \frac{p(B \cap Y)}{p(Y)} \\ &= p(B \mid Y) \end{aligned}$$

By the derivation above, we proved the given equation.

(f) Answer:
Let Y be "Baltimore," A be "car is actually blue," and B be "Claimed the car is blue."
Then the original equation becomes:

$$\begin{aligned} p(\text{car is actually blue} \mid \text{Claimed the car is blue}, \text{Baltimore}) &= \frac{p(B \mid A, Y) \cdot p(A \mid Y)}{p(B \mid A, Y) \cdot p(A \mid Y) + p(B \mid \bar{A}, Y) \cdot p(\bar{A} \mid Y)} \\ &= \frac{0.8 \times 0.1}{0.8 \times 0.1 + 0.2 \times 0.9} \\ &= 0.3077 \end{aligned}$$

Question 3.
(a) Answer:

$$\sum_{c \in cry} p(c \mid s) = 1, \text{ where } s \in \textit{situation}$$

(b) Answer:

p(cry, situation)	Predator!	Timber!	I need help!	TOTAL
bwa	0	0	0.64	0.64
bwee	0	0	0.08	0.08
kiki	0.2	0	0.08	0.28
TOTAL	0.2	0	0.8	1

(c) Answer:
i. This probability is written as: $p(\text{Predator} \mid \text{kiki})$
ii. It can be rewritten without the $\mid$ symbol as: $\frac{p(\text{Predator,kiki})}{p(\text{kiki})}$
iii. Using the above tables, its value is: $\frac{0.2}{0.28} = 0.7143$
iv. Alternatively, Bayes's Theorem allows you to express this probability as:

$$\frac{p(\text{kiki} \mid \text{Predator}) \cdot p(\text{Predatory})}{p(\text{kiki} \mid \text{Predator}) \cdot p(\text{Predator}) + p(\text{kiki} \mid \text{Timber}) \cdot p(\text{Timber}) + p(\text{kiki} \mid \text{I need help}) \cdot p(\text{I need help})}$$

v. Using the above tables, the value of this is:

$$\frac{1 \times 0.2}{1 \times 0.2 + 0 \times 0 + 0.1 \times 0.8} = 0.7143$$

Question 4.
(a) Answer:

$$p(\vec{w}) = \frac{c(BOS\ BOS\ w_1)}{c(BOS\ BOS)} \frac{c(BOS\ w_1\ w_2)}{c(BOS\ w_1)} \frac{c(w_1\ w_2\ w_3)}{c(w_1\ w_2)} ... \frac{c(w_{n-2}\ w_{n-1}\ w_n)}{c(w_{n-2}\ w_{n-1})} \frac{c(w_{n-1}\ w_n\ EOS)}{c(w_{n-1}\ w_n)}$$

$c(BOS\ BOS)$ counts the total number of sentences in the corpus.
$c(BOS\ BOS\ i)$ counts the total number of sentences that start with a word i in the corpus.
$c(new\ york\ EOS)$ counts the total number of sentences that end with *new york* in the corpus.

(b) Answer:
The probability of $< s >$ do you think the $< /s >$ should be extremely low because "the" has very low probability, or even close to zero, to be the last word in a sentence.

In the trigram model, the parameter that makes this probability low is $c(think\ the\ < /s >)$ because the "*think the* $< /s >$" will not happen many times in the corpus. *the*, a determiner, should be followed by a noun or noun phrase. But in the example, *the* is followed by $< /s >$.

(c) Answer:
Expression (A) is matched with description (2) because (A) doesn't have $< s >$ or $< /s >$, we don't know whether it is a sentence. We only know that we hear the three words, "Do you think".

Expression (B) is matched with description (1) since (B) has both the $< s >$ and $< /s >$ to indicate the start and end mark of the sentence "Do you think".

Expression (C) is matched with description (3) because (C) only has $< s >$ indicating the start of a sentence but no $< /s >$. So we only can be sure that "Do you think" is the starting words of a sentence.

$p(\vec{w})$ is to calculate the probability of the sentence using trigram model. Since a sentence needs a $< s >$ and a $< /s >$, quantity (B) is suitable for describing the trigram model probability of a sentence.

(d) Answer:
We first expand the equation for both $p(w)$ and $p_{reversed}(\vec{w})$ and show they are the same.
For $p(\vec{w})$

$$\begin{aligned} p(\vec{w}) &= \frac{p(w_1, w_0, w_{-1})}{p(w_0, w_{-1})} \frac{p(w_2, w_3, w_4)}{p(w_3, w_4)} ... \frac{p(w_n, w_{n-1}, w_{n-2})}{p(w_{n-1}, w_{n-2})} \frac{p(w_{n+1}, w_n, w_{n-1})}{p(w_n, w_{n-1})} \\ &= \frac{p(w_1, BOS, BOS)}{p(BOS, BOS)} \frac{p(w_2, w_3, w_4)}{p(w_3, w_4)} ... \frac{p(w_n, w_{n-1}, w_{n-2})}{p(w_{n-1}, w_{n-2})} \frac{p(EOS, w_n, w_{n-1})}{p(w_n, w_{n-1})} \end{aligned}$$

If we use "i love new york" as an example for this model, then it becomes

$$p(\text{i love new york}) = \frac{p(i, BOS, BOS)}{p(BOS, BOS)} \frac{p(love, i, BOS)}{p(i, BOS)} \frac{p(new, love, i)}{p(love, i)} \frac{p(york, new, love)}{p(new, love)} \frac{p(EOS, york, new)}{p(york, new)} \tag{4.d.1}$$

For $p_{reversed}(\vec{w})$ (notice that w_1 is the right most word in the origin sentence)

$$\begin{aligned} p_{reversed}(\vec{w}) &= \frac{p(w_0, w_1, w_2)}{p(w_1, w_2)} \frac{p(w_1, w_2, w_3)}{p(w_2, w_3)} ... \frac{p(w_{n-1}, w_n, w_{n+1})}{P(w_n, w_{n+1})} \frac{p(w_n, w_{n+1}, w_{w+2})}{p(w_{n+1}, w_{n+2})} \\ &= \frac{p(BOS, w_1, w_2)}{p(w_1, w_2)} \frac{p(w_1, w_2, w_3)}{p(w_2, w_3)} ... \frac{p(w_{n-1}, w_n, w_{n+1})}{P(w_n, w_{n+1})} \frac{p(w_n, EOS, EOS)}{p(EOS, EOS)} \end{aligned}$$

If we use "i love new york" as an example for this model, then it becomes

$$p_{reversed}(\text{i love new york}) = \frac{p(BOS, york, new)}{P(york, new)} \frac{p(york, new, love)}{p(new, love)} \frac{p(new, love, i)}{p(love, i)} \frac{p(love, i, EOS)}{p(i, EOS)} \frac{p(i, EOS, EOS)}{p(EOS, EOS)} \tag{4.d.2}$$

Since the "EOS" in $p(\vec{w})$ equals to the "BOS" in $p_{reversed}(\vec{w})$ and the "BOS" in $p(\vec{w})$ equals to the "EOS" in $p_{reversed}(\vec{w})$, the equation (4.d.1) equals to (4.d.2). If we observe the two equations, they have the same structure. Thus we have proved that $p(\vec{w}) = p_{reversed}(\vec{w})$.

Question 5.
Answer:
We added "a" denoting the topic variable and "h" denoting word history. Then make the language model condition on topic and word history.

$$p(w \mid h) = \sum_a p(w \mid a) p(a \mid h)$$

Next, we rewrite it as a bigram language model.

$$p(w \mid h) = \sum_a \prod_{i=1}^{n} p(w_i \mid w_{i-1}, a) p(a \mid \text{words before } w_i)$$

Based on the modified model, the formula for $p(w_1 w_2 w_3 w_4)$ can be written as

$$\begin{aligned} p(w \mid h) &= p(w_1 w_2 w_3 w_4 \mid h) \\ &= \sum_a p(w_1 \mid < s >, a) p(a \mid < s >) p(w_2 \mid w_1, a) p(a \mid < s >, w_1) ... p(< /s > \mid w_n, a) p(a \mid < s >, w_1, ..., w_n) \end{aligned}$$

To simply the computation, we can apply backoff on the history to make it depend on only at most some certain number of words in the begining of a sentence. The proposed model will become as below (we take first three words, including $< s >$ in the sentence into account):

$$\begin{aligned} p(w \mid h) &= p(w_1 w_2 w_3 w_4 \mid h) \\ &= \sum_a p(w_1 \mid < s >, a) p(a \mid < s >) p(w_2 \mid w_1, a) p(a \mid < s >, w_1) ... p(< /s > \mid w_n, a) p(a \mid < s >, w_1, w_2) \end{aligned}$$

To make this equation easier, we could only consider the probability of each topic in the training corpus. Thus, the model becomes

$$p(w) = \sum_a \prod_{i=1}^{n} p(w_i \mid w_{i-1}, a) p(a)$$

Question 8.
(a) Answer:
The top 5 most similar words to **seattle** are:

Ranking	Word	Cosine Similarity
1	seahawks	0.758477017352
2	spokane	0.753791653178
3	tacoma	0.713077987387
4	florida	0.710194673562
5	atlanta	0.684451194731

The top 5 most similar words to **dog** are:

Ranking	Word	Cosine Similarity
1	badger	0.827403953553
2	dogs	0.799978724722
3	hound	0.799708354606
4	cat	0.79232823091
5	borzoi	0.765538313797

The top 5 most similar words to **communist** are:

Ranking	Word	Cosine Similarity
1	socialist	0.874821580314
2	communists	0.818631237596
3	comintern	0.812112934282
4	bolshevik	0.794996023603
5	leftist	0.78247516237

The top 5 most similar words to **jpg** are:

Ranking	Word	Cosine Similarity
1	png	0.757922362133
2	svg	0.658103171326
3	galleria	0.634697828169
4	gif	0.614576269941
5	fuji	0.609729535714

The top 5 most similar words to **the** are:

Ranking	Word	Cosine Similarity
1	its	0.783370512889
2	in	0.770665320002
3	entire	0.764974819916
4	of	0.752080748141
5	which	0.742970631173

The top 5 most similar words to **google** are:

Ranking	Word	Cosine Similarity
1	com	0.745915499849
2	yahoo	0.73721515423
3	faq	0.726275514433
4	flickr	0.697346988155
5	web	0.689164493716

The top 5 most similar words to **baltimore** are:

Ranking	Word	Cosine Similarity
1	colts	0.715448935471
2	unitas	0.715169266716
3	philadelphia	0.711700587058
4	dallas	0.673996636042
5	cleveland	0.67395991933

The top 5 most similar words to **go** are:

Ranking	Word	Cosine Similarity
1	get	0.804380163332
2	going	0.791917570246
3	wait	0.767667283823
4	want	0.735100324766
5	leave	0.7046638208

From the examples shown above, "communist" worked best because "findsim" found the words that are similar to "communist." "go", and "the" worked poorly because the program returned words that are not

similar to the them regarding their semantic. For example, "entire" is not similar to "the," and "wait" is not similar to "go" either. Other examples worked well because even the words are not similar to the target words regarding semantic property, those words are related to the target words. For instance, "cat" and "dog" are both animals.

If we use smaller d, the result become worse. Take "jpg" for example, see the following result based on 10-dimension vector. The result shows less similar words than the previous result of "jpg."
The top 5 most similar words to **jpg** are:

Ranking	Word	Cosine Similarity
1	wan	0.93912896681
2	maui	0.935697578676
3	crannogs	0.93356830967
4	bahnhof	0.928524004063
5	tor	0.923776536092

On the other hand, if we use higher d, some results become better. For instance, the following result of "baltimore" is based on 200-dimension vector. Although those word are not similar to "balitmore" regarding semantic property, they are related to "baltimore." "colts" is the former football team based on Baltimore and "irsay" is the owner of the team "colts." In addition, "maryland" is more relevant to "baltimore" than "cleveland", "dallas", and "philadelphia" do.

The top 5 most similar words to **baltimore** are:

Ranking	Word	Cosine Similarity
1	maryland	0.552843128503
2	irsay	0.513162445035
3	hagerstown	0.491446562888
4	colts	0.483818773756
5	philadelphia	0.467494981195

(b) Answer:

The top 5 most similar words to **king - man + woman** are:

Ranking	Word	Cosine Similarity
1	queen	0.601229954946
2	betrothed	0.511752594404
3	consort	0.50889281754
4	heiress	0.506015397295
5	daughter	0.49657862759

The top 5 most similar words to **paris - france + uk** are:

Ranking	Word	Cosine Similarity
1	london	0.451385461104
2	odeon	0.440057066559
3	manchester	0.433118122934
4	promo	0.424648134466
5	molview	0.385688566984

The top 5 most similar words to **hitler - germany + italy** are:

Ranking	Word	Cosine Similarity
1	mussolini	0.534804309489
2	speer	0.427586316995
3	eichmann	0.419265599542
4	petacci	0.417031583213
5	graziani	0.416849390689

The top 5 most similar words to **child - goose + geese** are:

Ranking	Word	Cosine Similarity
1	children	0.515179273254
2	parents	0.460154553343
3	infants	0.456994574398
4	parenting	0.424722392905
5	infanticide	0.42345248145

The top 5 most similar words to **goes - eats + ate** are:

Ranking	Word	Cosine Similarity
1	went	0.562417091534
2	going	0.520057339135
3	go	0.494516630298
4	got	0.48438702187
5	forgot	0.436886429364

The top 5 most similar words to **car - road + air** are:

Ranking	Word	Cosine Similarity
1	aircraft	0.492352479725
2	pressurized	0.477225898754
3	helicopter	0.476701914758
4	parachutes	0.472783841942
5	bomber	0.456567849462

The top 5 most similar words to **fish - water + air** are:

Ranking	Word	Cosine Similarity
1	airbase	0.45353525027
2	marine	0.444535774596
3	corsairs	0.438252257403
4	warbird	0.43731466043
5	squadrons	0.434321217474

The top 5 most similar words to **baseball - music + guitar** are:

Ranking	Word	Cosine Similarity
1	mlb	0.498584908014
2	slugger	0.496346693662
3	lofton	0.486907223076
4	manny	0.483677369284
5	alou	0.478289071071

The following results are based on 10-dimension vector.

The top 5 most similar words to **fish - water + air** are:

Ranking	Word	Cosine Similarity
1	mills	0.944182264876
2	hmas	0.925994377082
3	scout	0.905816290242
4	hanford	0.898035552892
5	superfund	0.896362367403

The top 5 most similar words to **baseball - music + guitar** are:

Ranking	Word	Cosine Similarity
1	mvps	0.958718104671
2	player	0.949188663523
3	ace	0.945560961722
4	yount	0.94276522126
5	heisman	0.936004993904

This method for finding analogy works well for ”paris - france + uk”, ”king - man + woman”, ”car - road

+ air", and "baseball - music + guitar." However, if we use $d = 10$ instead of $d = 200$, it didn't work well for some case. For example, it found less similar words for "fish - water + air".

This method of finding analogies will find syntactic and semantic similar words or relative words. For example, it found "colts", which is a football team based on Baltimore, for the word "baltimore". It is not similar regarding syntactic or semantic aspects (or we can say they are both a noun).

The role of the vector "king - man" before "woman" is added means that a word has the property of "king" but no property of "man." So, after "man" was subtracted from "king," we might have the word that have a property like the leader of a group. After we added "woman" into the vector, we might have the word that has the properties like a leader and a woman. So we got the word "queen."

This technique can solve analogies because the found word is similar to the vector "A - B + C." So, if we let $A - B + C = D$, where D is the analogy word, and move elements in the equation to make it become $A - D = B - C$. By the new equation, we can find a word D that is similar to A just like C is similar to B.

Question 9.
The chain rule of $p(A, B, C, D)$ is

$$p(A, B, C, D) = p(A \mid B, C, D)p(B \mid C, D)p(C \mid D)p(D)$$

Then we apply backoff on the equation:

$$\begin{aligned}
p(A, B, C, D) &= p(A \mid B, D)p(B \mid C)p(C \mid D)p(D) \quad \text{(assume each pixel only depends on adjacent pixels)} \\
&= \frac{p(A, B, D)}{p(B, D)} p(B \mid C)p(C \mid D)p(D) \\
&= \frac{p(D \mid A, B)p(A \mid B)p(B)}{p(B)p(D)} p(B \mid C)p(C \mid D)p(D) \\
&\qquad \text{(assume } B \text{ and } D \text{ are independent, and } A \text{ and } C \text{ are independent)} \\
&= p(D \mid A)p(A \mid B)p(B \mid C)p(C \mid D) \\
&= p(A \mid B)p(B \mid C)p(C \mid D)p(D \mid A)
\end{aligned}$$

Thus, we have proven that $p(A, B, C, D)$ can be approximated to $p(A \mid B)p(B \mid C)p(C \mid D)p(D \mid A)$ by using chain rule and backoff.

Question 10.
Answer:
We first convert the following probability using chain rule:

$p(\neg fortune, \neg race, \neg horse, \neg shoe \mid \neg nail)$
$= p(\neg fortune \mid \neg race, \neg horse, \neg shoe, \neg nail)p(\neg race \mid \neg horse, \neg shoe, \neg nail)p(\neg horse \mid \neg shoe, \neg nail)p(\neg shoe \mid \neg nail)$

Since $p(\neg fortune \mid \neg race) = 1$, $p(\neg fortune \mid \neg race, \neg horse, \neg shoe, \neg nail)$ will also be 1 by (1k) and so do $p(\neg race \mid \neg horse, \neg shoe, \neg nail)$ and $p(\neg horse \mid \neg shoe, \neg nail)$.

Hence, we know that

$$p(\neg fortune, \neg race, \neg horse, \neg shoe \mid \neg nail) = 1 \times 1 \times 1 \times 1 = 1$$

In addition, because $(\neg fortune \cap \neg race \cap \neg house \cap \neg shoe \cap \neg nail) \subseteq (\neg fortune)$, $p(\neg fortune, \neg race, \neg house, \neg shoe, \neg nail) \leq p(\neg fortune)$ by (1a). Combine this equation with the above equation and we have:

$$p(\neg fortune, \neg race, \neg house, \neg shoe, \neg nail) \leq p(\neg fortune) \leq 1$$

We further make both side condition on $\neg nail$, then we have the following equation and the probability after adding a condition is still between [0,1] by (1b):

$$p(\neg fortune, \neg race, \neg house, \neg shoe, \neg nail \mid \neg nail) \leq p(\neg fortune \mid \neg nail) \leq 1$$

Since we already know that $p(\neg fortune, \neg race, \neg horse, \neg shoe \mid \neg nail) = 1$, the above equation will become

$$1 \leq p(\neg fortune \mid \neg nail) \leq 1$$

Therefore, $p(\neg fortune \mid \neg nail)$ must also be 1.