Natural Language Processing – Homework 2

מגיש 1: עמית רוקח

תעודת זהות 1: 322853813

מגיש 2: גיא קוך

תעודת זהות 2: 318962909

Question 1

a. Let $CE(y, \hat{y}) = -\sum_i y_i * log(\hat{y}_i)$ be the cross-entropy loss function

$$\hat{y}_i = \text{Softmax}(\Theta)_i = \frac{\exp(\Theta_i)}{\sum_j \exp(\Theta_j)},$$
$$\hat{y} = predicted\ probability\ vector, y = one - hot\ vec.$$

We can use the fact that y is a one-hot vector and we get

$$CE(y, \hat{y}) = -\sum_i y_i * log(\hat{y}_i) = -log(\hat{y}_k)$$

Where k is the index of the true class.

By the chain rule we get:

$$\frac{\partial CE}{\partial \Theta_i} = -\frac{1}{\hat{y}_k} * \frac{\partial \hat{y}_k}{\Theta_i}$$

Note that if $i = k$:

$$\frac{\partial \hat{y}_k}{\Theta_i} = \hat{y}_k(1 - \hat{y}_k)$$

if $i != k$:

$$\frac{\partial \hat{y}_k}{\Theta_i} = -\hat{y}_k \hat{y}_i$$

Now if we multiply by the chain rule we get

$if\ i = k$:

$$\frac{\partial CE}{\partial \Theta_i} = -\frac{1}{\hat{y}_k} * \frac{\partial \hat{y}_k}{\Theta_i} = -\frac{1}{\hat{y}_k}\ \hat{y}_k(1 - \hat{y}_k) = \hat{y}_k - 1$$

$if\ i != k$:

$$\frac{\partial CE}{\partial \Theta_i} = -\frac{1}{\hat{y}_k} * \frac{\partial \hat{y}_k}{\Theta_i} = \frac{1}{\hat{y}_k} \hat{y}_k \hat{y}_i = \hat{y}_i$$

Since y is a one-hot vector we can also write it as:

$$\forall i. \frac{\partial CE}{\partial \Theta_i} = \hat{y}_i - y_i$$

And in a more general form:

$$\frac{\partial CE}{\partial \Theta} = \hat{y} - y$$

b. Let

$$r_1 = xW_1 + b_1$$

$$h = \sigma(r1)$$

$$r_2 = hW_2 + b_2$$

$$\hat{y} = softmax(r_2)$$

$$y\ is\ a\ one - hot\ vector$$

$$CE(y, \hat{y}) = -\sum_i y_i \log(\hat{y}_i)$$

$$J = CE(y, \hat{y})$$

We want to compute $\frac{\partial J}{\partial x}$.

Using the chain rule, we get:

$$\frac{\partial J}{\partial x} = \frac{\partial J}{\partial r_2}\frac{\partial r_2}{\partial h}\frac{\partial h}{\partial r_1}\frac{\partial r_1}{\partial x}$$

We will compute each of the derivatives:

$$\frac{\partial r_1}{\partial x} = \frac{\partial(xW_1 + b_1)}{\partial x} = W_1$$

$$\frac{\partial h}{\partial r_1} = \frac{\partial \sigma(r1)}{\partial r_1} = \sigma(r1) \odot \big(1 - \sigma(r1)\big) = h \odot (1 - h)$$

$$\frac{\partial r_2}{\partial h} = \frac{\partial hW_2 + b_2}{\partial h} = W_2$$

$$\frac{\partial J}{\partial r_2} = \hat{y} - y$$

$\odot$ - element wise multipication

Therefore, we get:

$$\frac{\partial J}{\partial x} = \Big(\big((\hat{y} - y)W_2^T\big) \odot \big(h \odot (1 - h)\big)\Big) W_1^T$$

d. We got a dev perplexity of 112.81714029.
saved_params_40000.npy is included in the zip file.

Question 2

a.

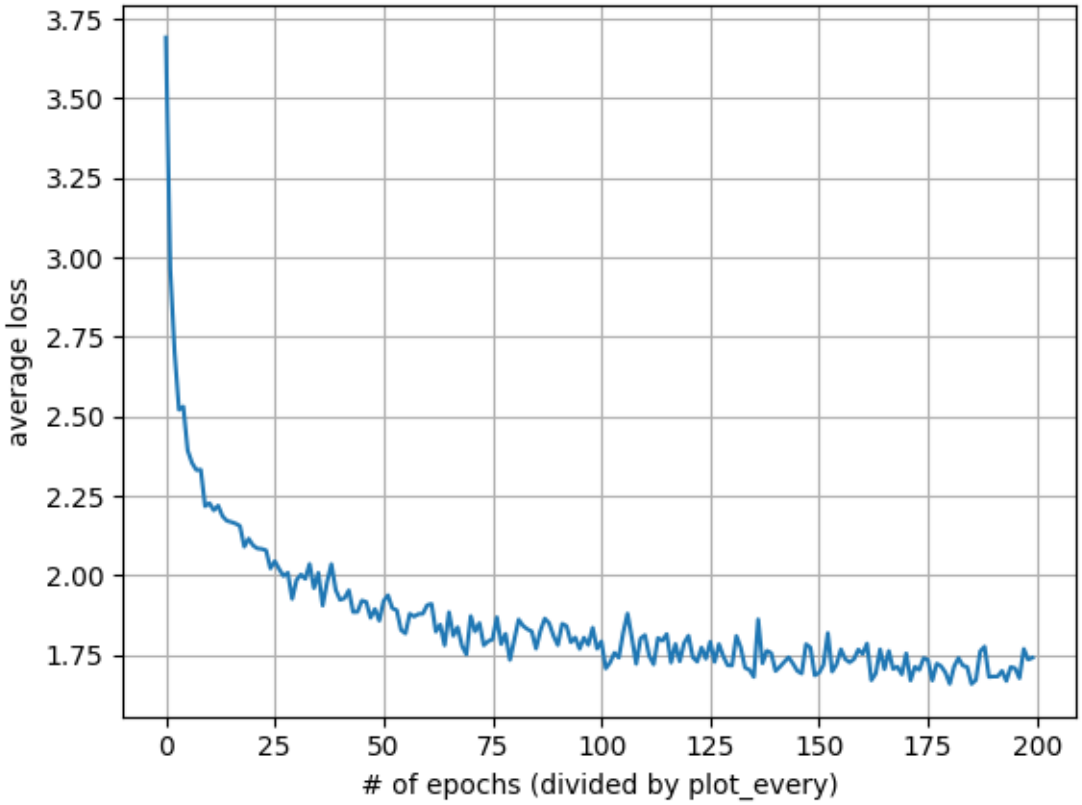
- Advantage of character based language model over word based language model:

  Character based language models' vocabulary is the alphabet or a small set of characters, which means that they are better in handling rare or unseen words (in comparison to word based language models). The latter is due to the model's ability to "learn" patterns in character sequences rather than just treating unknown words as a "new" word which is what word based language models do.

- Advantage of word based language model over character based language model:

  Since word based language models' vocabulary is a group of words, they have lower computational overhead. The latter is due to the fact that this group of models treat entire words as a single unit, which leads to them processing fewer tokens when processing text (in comparison to character based language models).

b. Losses plot:



Question 3

a. $2^{-\frac{1}{M}\sum_{i=1}^{M}\log_2 p(S_i|S_1,\dots,S_{i-1})} \underset{\log_a b=\frac{lnb}{lna}}{=} 2^{-\frac{1}{M}\sum_{i=1}^{M}\frac{1}{ln2}*ln\, p(S_i|S_1,\dots,S_{i-1})} =$
$2^{\frac{1}{ln2}*\left(-\frac{1}{M}\sum_{i=1}^{M} ln\, p(S_i|S_1,\dots,S_{i-1})\right)} = (2^{\frac{1}{ln2}})^{\left(-\frac{1}{M}\sum_{i=1}^{M} ln\, p(S_i|S_1,\dots,S_{i-1})\right)} =$
$e^{-\frac{1}{M}\sum_{i=1}^{M} ln\, p(S_i|S_1,\dots,S_{i-1})}$

b. **Results:**

Model from section 2:

- Perplexity of shakespeare_for_perplexity.txt: 7.1626771592940575.
- Perplexity of wikipedia_for_perplexity.txt: 18.425545474390916.

Code: can be found at the last code block in file "Copy_of_q3_char_rnn_generation.ipynb" right under the text that says "Q3".

Bi-gram lm:

- Perplexity of shakespeare_for_perplexity.txt: 7.50448069.
- Perplexity of wikipedia_for_perplexity.txt: 30.17933101.

Code: can be found in the main of file "q1d_neural_lm.py".

c. We will explain the results for each model, while focusing on the large gaps in perplexity for the different passages:
   - We think the reason for the large gap in perplexity for the "Model from section 2" between the Shakespeare text file and the Wikipedia text file is due to the reason the model was trained on the Shakespeare dataset which might have helped it capture the consistent character patterns (since its a character-based lm) and syntax unique to Shakespeare's writing.
   - We think the reason for the large gap in perplexity for the bi-gram language model between the Shakespeare text file and the Wikipedia text file is due to the Wikipedia text file being very concise about New Zealand and might be repeating words that did not appear in the training set, while the Shakespeare text file might have less repeating words that did not appear in the training set.

d. We decided on doing two different preprocessing, one for each model.
   We will now describe the preprocessing done for the "Model from section 2":
   1. Remove punctuation from the text, leaving only alphanumeric characters and whitespace.
   2. Convert the entire text to lowercase to make it case-insensitive.

   Results after preprocessing for the "Model from section 2" language model:

   - Perplexity of preprocessed shakespeare_for_perplexity.txt: 7.9803277217625475.

- Perplexity of preprocessed wikipedia_for_perplexity.txt: 15.31237554767049.

We can see that in this case the perplexity has improved for the Wikipedia preprocessed text file, but it came with a trade-off of a worse perplexity for the Shakespeare preprocessed text file. We see that as an improvement since it reduces the gap in perplexity between the different files and the improvement in favor of the Wikipedia preprocessed text file was significant while the decline (since the perplexity went up in comparison to the original text file) in the perplexity of the Shakespeare preprocessed text file was pretty mild.

We will now describe the preprocessing done for the bi-gram language model:

1. Remove punctuation from the text, leaving only alphanumeric characters and whitespace.
2. Convert the entire text to lowercase to make it case-insensitive.
3. Split the text into individual words based on spaces.
4. Create bigrams by pairing each consecutive word in the list.
5. Convert the bigrams into a string format where each bigram is separated by a newline.

Results after preprocessing for the bi-gram language model:

- Perplexity of preprocessed shakespeare_for_perplexity.txt: 7.06231863.
- Perplexity of preprocessed wikipedia_for_perplexity.txt: 7.01997949.

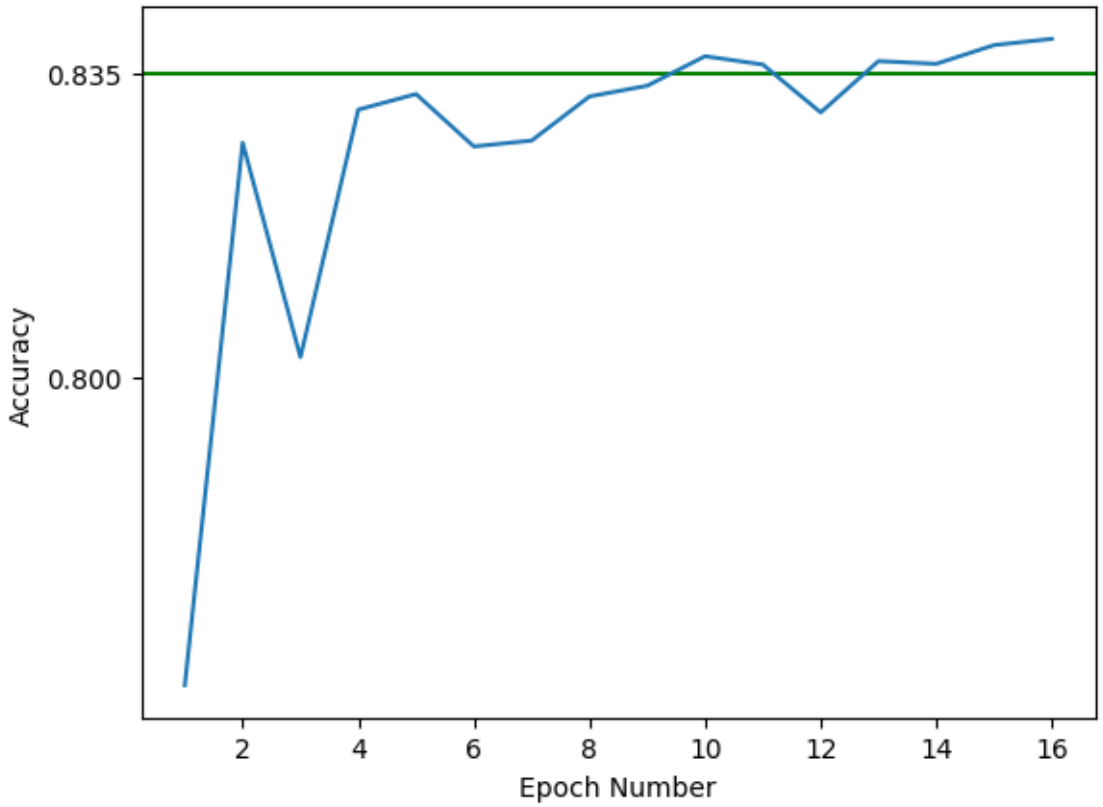
We can see that in this case the gap in perplexity between the different text files has decreased significantly. Also, we can see that the perplexity has improved for both preprocessed test files while the one that has improved significantly more is the preprocessed Wikipedia text file.

* The preprocessing was done using new functions added to the py/ipynb files:

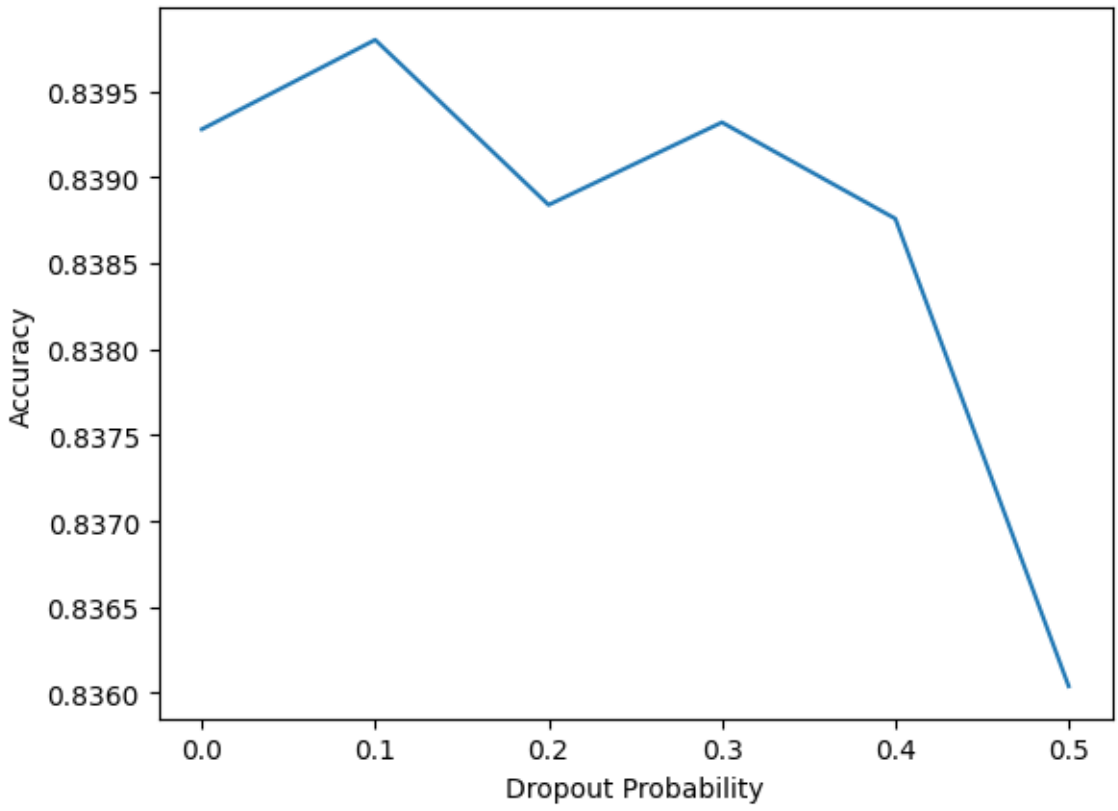
- For the “Copy_of_q3_char_rnn_generation.ipynb” file we added the “perplexity_with_preprocessing” function.
- For the “q1d_neural_lm.py” we added the following 4 functions:
  - eval_neural_lm_with_preprocessing – in the file itself.
  - load_data_as_sentences_with_preprocessing – in the file itself.
  - load_dataset_with_preprocessing – in data_utils/utils.py.
  - preprocess (used by load_dataset_with_preprocessing) – in data_utils/utils.py.

Question 4

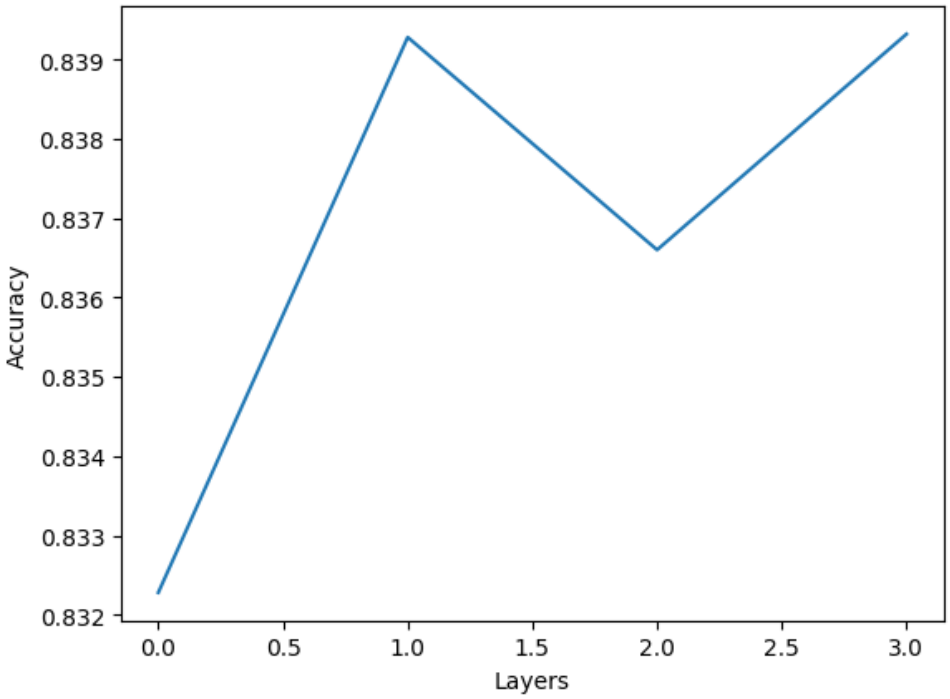
a. A plot of the evaluation accuracy as a function of the number of epochs:



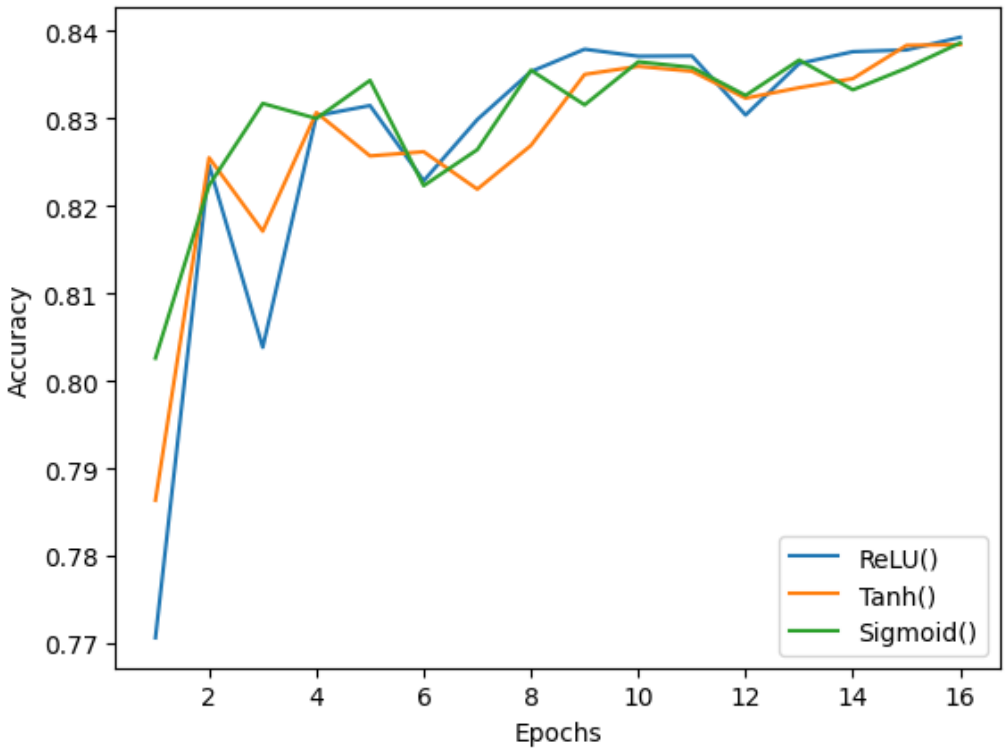
b. A plot of the accuracy of the model across different values of the dropout rate:



c. We thought that we will start seeing the effect of diminishing returns after 1 hidden layer.
As we expected, We do not see any significant improvement by adding more than 1 hidden layer.
The linear model has not outperformed the model with 3 hidden layers (in the forum we were told we can compare it to the model with 3 hidden layers instead of 4).
Plot of the accuracy as a function of the number of layers:



d. A plot of the accuracy across epochs for the different activation functions:



We have learnt from this experiment that the activation function does not have a significant effect on the accuracy of the model across the epochs (at least for the configuration we used).
e. The following are 5 examples from the evaluation set that the model classified incorrectly:

1. **Example:** I gave 9 of 10 points. I was sitting in tears nearly the whole movie, because I had to laugh!

The story of course wasn\'t excellent, but it also wasn\'t boring. Erkan & Stefan are assigned to become bodyguards for the beautiful Nina. While doing this job they come between the "front-lines" of BND and CIA. Of course the two are neither born bodyguards nor gentlemen, so they run from one disaster into another; and they do this in such a funny way, that when you watch some scenes you won\'t be able to stop the tears! As actors those two "dumbly grinning" characters do quite well, better than some so called professional.

You think, the speech of the two heroes is curios or "pseudo-foreign"? Well, if you hear quite a lot Turkish-German people in Munich speaking exactly like them, you will remember Erkan & Stefan. And maybe, in 10 years it might have become the common speech of the youth. (God forbid!)

So, if you like to laugh, watch this movie!
   Label: 1
   Model Prediction: 0
   Why The Model Classified It Incorrectly: The model likely predicted 0 because the review contains phrases like "The story of course wasn\'t excellent" and "God forbid!" which may have been interpreted as negative sentiment.
2. **Example:** I love this movie a lot. I must get this on DVD. I have 2 VHS copies, but the quality is so poor that you can\'t read one written joke over the door of the ward. I\'m forever amazed that Blankfield did almost nothing afterward. He made both Dr. Jeckle and Mr. Hyde totally believable.

The movie is plagued by it\'s low budget. (One atrocious edit jumps into mid-word and was described on, "Siskel & Ebert".) But, there are a thousand jokes, sight gags to subtle references, that more than compensate. I often find myself quoting lines (or, singing, "I\'ve Got Nothing to Hide") and, from time to time, completely describe a scene which matches some conversation. There are, at least, six scenes which are among my all time favorite comedy bits.

Viewers with no history of cocaine use may miss a lot of gags.

"Here, take it." * Visual of driving while waving butt out the window.* "I said, \'Is this seat taken?\'" "Nice Burn!" Visual of chaps, headdress, jockstrap, & swim fins. * "Yeah. I\'m right handed." * "Me! Me!" says the woman trying to sell \'nads. * "Bernie\'s going to love these." * "That\'s my feet, Jack." says the black feet. * "Why should we tell you?"... "SHE\'S AT THE SUPERMARKET!" * "Ivy!" on supermarket PA. * Loading whole shopping cart into ambulance. * etc.
   Label: 1

Model Prediction: 0
Why The Model Classified It Incorrectly: The model likely predicted 0 due to phrases highlighting flaws like " The movie is plagued by it\'s low budget" and "one atrocious edit" which may have been interpreted as negative sentiment.

3. **Example:** One of the last surviving horror screen greats - Conrad Radzoff - dies and has his body placed in a mausoleum with televised-before-death snippets of the great Conrad greeting you as you visit. Unfortunately for him and his captors, Conrad\'s body is "borrowed" by a gang of four boys and three girls and taken to a huge manor where they drink with him, toast him, dance with him, laugh with and at him, and then put him to bed in a casket which just happens to by lying in a room upstairs. News of the missing body reaches Radzoff\'s widow and her friend(who happens to be proficient in the black arts) and she holds some kind of ceremony that brings Conrad back to life so he can, in his own words, get "an eye for an eye, a tooth for a tooth." Well, Frightmare is an interesting "bad" film. Sure, it is cheap. The sets look like they were borrowed(which I am sure they were). The special effects and blood and guts are done liberally and with little credibility. The acting is average to below average with a few exceptions. Jeffrey Combs of Re-Animator fame is in tow, but really he does little in this rather thankless role as a horror obsessed teen that needs to steal a dead man\'s body for kicks. None of the "kids" except the pretty girl playing Meg is any good. Nita Talbot plays the "friend" of the Radzoffs with withering interest. Also, look for the big - I mean big - guy that plays the policeman. That is Porky himself of Porkys fame. But thankfully for all of us, one performance does rise above the material. Ferdy Mayne, an oft overlooked actor from Germany who had Christopher Lee features and did star as a vampire in The Fearless Vampire Killers, does a more than commendable job as the aging horror icon in public life and a real demon of a man in private life. Conrad Radzoff in a bad human being in life, living solely for his own pleasures and we see him kill twice before he is even dead(obviously none of the swinging teens at that point). Mayne is able to look very regal, speak very elegantly, and convey menace with ease. If for no other reason, one should see Frightmare for his performance. I do; however, believe that when they showed black and white clips of Radzoff that they used Christopher Lee footage(anyone have any thoughts?). Anyway, one can guess what happens and it does indeed: Radzoff goes out and goes after the kids that disturbed his peace. Again, the formula is trite and overused. The acting for the most part is anemic, and the direction oh so ridiculous. But Mayne gives a good performance in a sea of ineptitude. Definitely worth a little peek. Watching Mayne keep popping up on screens in his mausoleum brought a wry smile to my lips each time.
Label: 0

Model Prediction: 1
Why The Model Classified It Incorrectly: The model likely predicted 1 because of phrases like "definitely worth a little peek", "Mayne gives a good performance", and "brought a wry smile to my lips" which may have been interpreted as positive sentiment.

4. **Example:** This is the funniest sequel I have seen in a long time it is much funnier than the other three and not a bit scary. It has some very gory pieces in the film, but not bad enough to make you sick. In this one he has a female doll companion, hence the name. If you liked the first three then you'll love this, go watch it!
   Label: 1
   Model Prediction: 0
   Why The Model Classified It Incorrectly: The model likely predicted 0 because phrases like "not a bit scary" and "some very gory pieces" which may have been interpreted as negative sentiment.
5. **Example:** I bought this DVD as part of a set of 50 "historic classics." It\'s hardly a classic, and as the plot was updated to the time of its release, is not historic either. The actual title on the DVD is "Indecent," and additionally subtitled "The Private Life of Becky Sharp." Myrna Loy is not very convincing, although in her defense she is saddled with an awful script and trite dialogue. As with many early talkies, and especially ones made by smaller studios, there is little skill demonstrated by the cast and crew. Loy does wear a few gowns that are quite stylish, but her costumes and make-up in the later scenes are overdone. The one saving grace is a tolerable performance by Billy Bevan, who plays one of her many suitors
   Label: 0
   Model Prediction: 1
   Why The Model Classified It Incorrectly: The model likely predicted 1 because of phrases like "Loy does wear a few gowns that are quite stylish" and "a tolerable performance by Billy Bevan" which could have been interpreted as positive.

Question 5

1.

a. α can be interpreted as a categorical probability distribution due to the following reasons:
   1. $For\ all\ i. \alpha_i \geq 0$ – since for every t, $\exp(t) \geq 0$ and α's definition.
   2. $\sum_{i=1}^{n} \alpha_i = 1$ –
   $$\sum_{i=1}^{n} \alpha_i = \sum_{i=1}^{n} \frac{\exp\left(k_i^T q\right)}{\sum_{j=1}^{n} exp(k_j^T q)} = \frac{\sum_{i=1}^{n} \exp(k_i^T q)}{\sum_{j=1}^{n} \exp(k_j^T q)} = 1$$
   3. Interpretation – Each α_i represents the weight assigned to the corresponding value v_i (category).

b. From $\alpha' s$ definition, for the categorical distribution α to put almost all its weight on a specific α_j, it must be true that: $k_j^T q \gg k_i^T q\ for\ all\ i \neq j$.

c. From the fact that $\sum_{i=1}^{n} \alpha_i = 1$, $For\ all\ i\ \alpha_i \geq 0$ and the condition from section b we get that $\alpha_j \approx 1$ and $For\ all\ i \neq j. \alpha_i \approx 0$ which leads to the fact that:
$$c = \sum_{i=1}^{n} \alpha_i v_i \approx 1 * v_j = v_j \rightarrow c \approx v_j$$

d. Intuitively, it means that our model will give $\approx v_j$ attention to the j word because it finds it more relevant to the query q compared to the other words.

2.a. We define A' to be the following matrix: $A' = [a_1\ a_2\ \dots a_n]$.

We define M to be $M = A'A'^T$.

Since the vectors $\{a_1, a_2, \dots, a_n\}$ are orthonormal we get that M is the projection matrix to the sub space A where v_a lies, thus:

$$Mv_a = v_a$$

From the fact that $a_j^T b_k = 0\ for\ all\ j, k$ and M being the projection matrix to the sub space A, we get that:

$$Mv_b = 0$$

We finally get:

$$Ms = M(v_a + v_b) = Mv_a + Mv_b = v_a + 0 = v_a$$

b. We select $q = G(k_a + k_b)$ where G is a large scalar, specifically large enough for the following condition to hold: $\exp(G) \gg n\ and \exp(G) \gg 1$.

$$\alpha_a = \frac{\exp(k_a^T q)}{\sum_{j=1}^{n} \exp\left(k_j^T q\right)} = \frac{\exp\left(k_a^T G(k_a + k_b)\right)}{\sum_{j=1}^{n} \exp\left(k_j^T G(k_a + k_b)\right)} \underset{For\ all\ i \neq j. k_j^T k_i = 0}{=}$$
$$= \frac{\exp(G k_a^T k_a + 0)}{\sum_{j=1\ AND\ j \neq a,b}^{n} \exp(0) + \exp(G k_a^T k_a + 0) + \exp(G k_b^T k_b + 0)} \underset{For\ all\ i. k_i^T k_i = 1}{=}$$

$$= \frac{\exp(G)}{n-2+2*\exp(G)} \approx \frac{\exp(G)}{2*\exp(G)} = \frac{1}{2}$$

Similarly, we can get that $\alpha_b \approx \frac{\exp(G)}{2*\exp(G)} = \frac{1}{2}$.

For every $i \neq a, b$ we get:

$$\alpha_i = \frac{\exp(k_i^T q)}{\sum_{j=1}^{n} \exp(k_j^T q)} = \frac{\exp\left(k_i^T G(k_a + k_b)\right)}{\sum_{j=1}^{n} \exp(k_j^T G(k_a + k_b))} \underset{For\ all\ l \neq j . k_j^T k_l = 0}{=}$$
$$= \frac{\exp(0)}{n-2+2*\exp(G)} = \frac{1}{n-2+2*\exp(G)} \approx 0$$

Now we calculate c:

$$c = \sum_{i=1}^{n} v_i \alpha_i = \sum_{j=1\ AND\ j \neq a,b}^{n} \mathrm{v_j} \alpha_j + v_a \alpha_a + v_b \alpha_b \approx \sum_{j=1\ AND\ j \neq a,b}^{n} 0 + \frac{1}{2} v_a + \frac{1}{2} v_b$$
$$= \frac{1}{2}(v_a + v_b)$$

3.a. Since the covariance matrices are $\sum_i = \alpha I$ for vanishingly small α we get that for all $i \in \{1,2,\dots,n\}$. $k_i \approx \mu_i$.

We select $q = G(\mu_a + \mu_b)$ where G is a large scalar, specifically large enough for the following condition to hold: $\exp(G) \gg n\ and \exp(G) \gg 1$.

$$\alpha_a = \frac{\exp(k_a^T q)}{\sum_{j=1}^{n} \exp(k_j^T q)} \approx \frac{\exp(\mu_a^T q)}{\sum_{j=1}^{n} \exp(\mu_j^T q)} = \frac{\exp\left(\mu_a^T G(\mu_a + \mu_b)\right)}{\sum_{j=1}^{n} \exp(\mu_j^T G(\mu_a + \mu_b))} \underset{For\ all\ i \neq j . \mu_j^T \mu_i = 0}{=}$$
$$= \frac{\exp(G\mu_a^T \mu_a + 0)}{\sum_{j=1\ AND\ j \neq a,b}^{n} \exp(0) + \exp(G\mu_a^T \mu_a + 0) + \exp(G\mu_b^T \mu_b + 0)} \underset{For\ all\ i . \mu_i^T \mu_i = 1}{=}$$
$$= \frac{\exp(G)}{n-2+2*\exp(G)} \approx \frac{\exp(G)}{2*\exp(G)} = \frac{1}{2}$$

Similarly, we can get that $\alpha_b \approx \frac{\exp(G)}{2*\exp(G)} = \frac{1}{2}$.

For every $i \neq a, b$ we get:

$$\alpha_i = \frac{\exp(k_i^T q)}{\sum_{j=1}^{n} \exp(k_j^T q)} \approx \frac{\exp(\mu_i^T q)}{\sum_{j=1}^{n} \exp(\mu_j^T q)} = \frac{\exp\left(\mu_i^T G(\mu_a + \mu_b)\right)}{\sum_{j=1}^{n} \exp(\mu_j^T G(\mu_a + \mu_b))} \underset{For\ all\ l \neq j . \mu_j^T \mu_l = 0}{=}$$
$$= \frac{\exp(0)}{n-2+2*\exp(G)} = \frac{1}{n-2+2*\exp(G)} \approx 0$$

Now we calculate c:

$$c = \sum_{i=1}^{n} v_i \alpha_i = \sum_{j=1\ AND\ j \neq a,b}^{n} \mathrm{v_j} \alpha_j + v_a \alpha_a + v_b \alpha_b \approx \sum_{j=1\ AND\ j \neq a,b}^{n} 0 + \frac{1}{2} v_a + \frac{1}{2} v_b$$
$$= \frac{1}{2}(v_a + v_b)$$

b. We will explain the behaviour of k_a and $\forall i \neq a. k_i$ separately:

- k_a:
  Since α is vanishingly small and $||\mu_a|| = 1$ we get that the variability along μ_a's direction scales only with the distribution of a random factor d such that $d \sim N(1, 0.5)$ (meaning that d varies between 0.5 and 1.5). From the latter we can infer $k_a \approx d\mu_a$.
- $\forall i \neq a. k_i$:
  Since the covariance matrices are $\sum_i = \alpha I$ for vanishingly small α we get that $k_i \approx \mu_i$.

In part a we defined the q vector to be $q = G(\mu_a + \mu_b)$ where G is a large scalar, specifically large enough for the following condition to hold: $\exp(G) \gg n \; and \; \exp(G) \gg 1$.

Since $\forall i. \mu_i$ are orthogonal to each other and the approximation of k_i defined above we get that:

$$\forall i \neq a, b. k_i^T q = 0$$

$$k_a^T q \approx d\mu_a^T * G(\mu_a + \mu_b) = dG\mu_a^T \mu_a + dG\mu_a^T \mu_b = dG * 1 + 0 = dG$$

$$k_b^T q \approx \mu_b^T * G(\mu_a + \mu_b) = G\mu_b^T \mu_a + G\mu_b^T \mu_b = 0 + G = G$$

We will now calculate α:

$$\alpha_a = \frac{\exp(k_a^T q)}{\sum_{j=1}^{n} \exp(k_j^T q)} \approx \frac{\exp(dG)}{\sum_{j=1 \; AND \; j \neq a,b}^{n} \exp(0) + \exp(dG) + \exp(G)}$$

$$= \frac{\exp(dG)}{n - 2 + \exp(dG) + \exp(G)}$$

$$\approx \frac{\exp(dG)}{\exp(dG) + \exp(G)} \underset{We \; devide \; both \; the \; numerator \; and \; denominator \; by \; \exp(dG)}{=} \frac{1}{1 + \exp(G - dG)}$$

$$= \frac{1}{1 + \exp(G(1 - d))}$$

$$\alpha_b = \frac{\exp(k_b^T q)}{\sum_{j=1}^{n} \exp(k_j^T q)} \approx \frac{\exp(G)}{\sum_{j=1 \; AND \; j \neq a,b}^{n} \exp(0) + \exp(dG) + \exp(G)}$$

$$= \frac{\exp(G)}{n - 2 + \exp(dG) + \exp(G)}$$

$$\approx \frac{\exp(G)}{\exp(dG) + \exp(G)} \underset{We \; devide \; both \; the \; numerator \; and \; denominator \; by \; \exp(G)}{=} \frac{1}{\exp(dG - G) + 1}$$

$$= \frac{1}{\exp(G(d - 1)) + 1}$$

$$\forall i \neq a, b. \alpha_i = \frac{\exp(k_i^T q)}{\sum_{j=1}^{n} \exp(k_j^T q)} \approx \frac{\exp(0)}{\sum_{j=1 \; AND \; j \neq a,b}^{n} \exp(0) + \exp(dG) + \exp(G)} \approx 0$$

So, we get that $c = \sum_{i=1}^{n} v_i \alpha_i \approx \alpha_a v_a + \alpha_b v_b$.

As we said before d varies between 0.5 and 1.5. We will now separate to different cases based on d's value and calculate c:

- $d = 0.5$:

$$\alpha_a \approx \frac{1}{1+\exp\big(G(1-0.5)\big)} = \frac{1}{1+\exp(0.5G)} \underset{\exp(G)\gg 1}{\approx} 0$$

$$\alpha_b \approx \frac{1}{\exp(G(0.5-1))+1} = \frac{1}{1+\exp(-0.5G)} \underset{\exp(G)\gg 1 \to \exp(-G)\approx 0}{\approx} 1$$

$$\boldsymbol{c \approx \alpha_a v_a + \alpha_b v_b \approx v_b}$$

- $d = 1.5$:

$$\alpha_a \approx \frac{1}{1+\exp\big(G(1-1.5)\big)} = \frac{1}{1+\exp(-0.5G)} \underset{\exp(G)\gg 1 \to \exp(-G)\approx 0}{\approx} 1$$

$$\alpha_b \approx \frac{1}{\exp(G(1.5-1))+1} = \frac{1}{1+\exp(0.5G)} \underset{\exp(G)\gg 1}{\approx} 0$$

$$\boldsymbol{c \approx \alpha_a v_a + \alpha_b v_b \approx v_a}$$

So we got that when $d \to 0.5$ then $c \approx v_b$ and when $d \to 1.5$ then $c \approx v_a$ which means that moves between v_b and v_a.

4.a. We select $q_1 = G\mu_a$ and $q_2 = G\mu_b$ where G is a large scalar, specifically large enough for the following condition to hold: $\exp(G) \gg n \; and \; \exp(G) \gg 1$.

We will first focus on q_1:

$$\alpha_{a_1} = \frac{\exp(k_a^T q_1)}{\sum_{j=1}^{n}\exp\,(k_j^T q_1)} \approx \frac{\exp(\mu_a^T q_1)}{\sum_{j=1}^{n}\exp\,(\mu_j^T q_1)} = \frac{\exp(\mu_a^T G\mu_a)}{\sum_{j=1}^{n}\exp\,(\mu_j^T G\mu_a)} \underset{For\ all\ i\neq j.\mu_j^T\mu_i=0}{=}$$

$$= \frac{\exp(G\mu_a^T\mu_a)}{\sum_{j=1\ AND\ j\neq a}^{n}\exp(0)+\exp(G\mu_a^T\mu_a)} \underset{For\ all\ i.\mu_i^T\mu_i=1}{=}$$

$$= \frac{\exp(G)}{n-1+\exp\,(G)} \approx \frac{\exp(G)}{\exp(G)} = 1$$

For every $i \neq a$ we get:

$$\alpha_{i_1} = \frac{\exp(k_i^T q_1)}{\sum_{j=1}^{n}\exp\,(k_j^T q_1)} \approx \frac{\exp(\mu_i^T q_1)}{\sum_{j=1}^{n}\exp\,(\mu_j^T q_1)} = \frac{\exp(\mu_i^T G\mu_a)}{\sum_{j=1}^{n}\exp\,(\mu_j^T G\mu_a)} \underset{For\ all\ l\neq j.\mu_j^T\mu_l=0}{=}$$

$$\frac{\exp(G\mu_i^T\mu_a)}{\sum_{j=1\ AND\ j\neq a}^{n}\exp(0)+\exp(G\mu_a^T\mu_a)} \underset{For\ all\ i.\mu_i^T\mu_i=1}{=} = \frac{\exp(0)}{n-1+\exp\,(G)} = \frac{1}{n-1+\exp\,(G)}$$

$$\approx 0$$

From here we get that:

$$c_1 = \sum_{i=1}^{n} v_i\alpha_i = \sum_{j=1\ AND\ j\neq a}^{n} \mathrm{v_j}\alpha_{j_1} + v_a\alpha_{a_1} \approx \sum_{j=1\ AND\ j\neq a}^{n} 0 + v_a = v_a$$

Similarly, for q_2, c_2 we get $\alpha_{b_2} \approx 1$, $\forall i \neq b.\, \alpha_{i_2} \approx 0$ which leads to $c_2 \approx v_b$.

Finally, we get:

$$c = \frac{1}{2}(c_1 + c_2) \approx \frac{1}{2}(v_a + v_b)$$

b. We will explain the behaviour of k_a and $\forall i \neq a. k_i$ separately:

- k_a:
  Since α is vanishingly small and $\left||\mu_a|\right| = 1$ we get that the variability along μ_a's direction scales only with the distribution of a random factor d such that $d \sim N(1, 0.5)$ (meaning that d varies between 0.5 and 1.5). From the latter we can infer $k_a \approx d\mu_a$.
- $\forall i \neq a. k_i$:
  Since the covariance matrices are $\sum_i = \alpha I$ for vanishingly small α we get that $k_i \approx \mu_i$.

In part a we defined $q_1 = G\mu_a$ and $q_2 = G\mu_b$ where G is a large scalar, specifically large enough for the following condition to hold: $\exp(G) \gg n \; and \; \exp(G) \gg 1$.

Since $\forall i. \mu_i$ are orthogonal to each other and the approximation of k_i defined above we get that:

- For q_1:

$$\forall i \neq a, b. k_i^T q_1 = 0$$

$$k_a^T q_1 \approx d\mu_a^T * G\mu_a = dG\mu_a^T\mu_a = dG * 1 = dG$$

$$k_b^T q_1 \approx \mu_b^T * G\mu_a = G\mu_b^T\mu_a = 0$$

- For q_2:

$$\forall i \neq a, b. k_i^T q_2 = 0$$

$$k_a^T q_2 \approx d\mu_a^T * G\mu_b = dG\mu_a^T\mu_b = 0$$

$$k_b^T q_2 \approx \mu_b^T * G\mu_b = G\mu_b^T\mu_b = G$$

We will now calculate c_1 and c_2:

- c_1:

$$\alpha_{a_1} = \frac{\exp(k_a^T q_1)}{\sum_{j=1}^{n} \exp\,(k_j^T q_1)} \approx \frac{\exp(dG)}{\sum_{j=1\; AND\; j\neq a}^{n} \exp(0) + \exp(dG)} = \frac{\exp(dG)}{n - 1 + \exp(dG)}$$
$$\approx \frac{\exp(dG)}{\exp(dG)} = 1$$

$$\alpha_{b_1} = \frac{\exp(k_b^T q_1)}{\sum_{j=1}^{n} \exp\,(k_j^T q_1)} \approx \frac{\exp(0)}{\sum_{j=1\; AND\; j\neq a}^{n} \exp(0) + \exp(dG)} = \frac{\exp(0)}{n - 1 + \exp(dG)} \approx 0$$

$$\forall i \neq a, b. \alpha_{i_1} = \frac{\exp(k_i^T q_1)}{\sum_{j=1}^{n} \exp\,(k_j^T q_1)} \approx \frac{\exp(0)}{\sum_{j=1\ AND\ j\neq a}^{n} \exp(0) + \exp(dG)} = \frac{\exp(0)}{n-1+\exp(dG)} \approx 0$$

So, we get that $c_1 = \sum_{i=1}^{n} v_i \alpha_{i_1} \approx 0 + v_a * 1 = v_a$.

- c_2:

$$\alpha_{a_2} = \frac{\exp(k_a^T q_2)}{\sum_{j=1}^{n} \exp\,(k_j^T q_2)} \approx \frac{\exp(0)}{\sum_{j=1\ AND\ j\neq b}^{n} \exp(0) + \exp(G)} = \frac{\exp(0)}{n-1+\exp(G)} \approx 0$$

$$\alpha_{b_2} = \frac{\exp(k_b^T q_2)}{\sum_{j=1}^{n} \exp\,(k_j^T q_2)} \approx \frac{\exp(G)}{\sum_{j=1\ AND\ j\neq b}^{n} \exp(0) + \exp(G)} = \frac{\exp(G)}{n-1+\exp(G)} \approx 1$$

$$\forall i \neq a, b. \alpha_{i_2} = \frac{\exp(k_i^T q_2)}{\sum_{j=1}^{n} \exp\,(k_j^T q_2)} \approx \frac{\exp(0)}{\sum_{j=1\ AND\ j\neq b}^{n} \exp(0) + \exp(G)} = \frac{\exp(0)}{n-1+\exp(G)} \approx 0$$

So, we get that $c_2 = \sum_{i=1}^{n} v_i \alpha_{i_2} \approx 0 + v_b * 1 = v_b$.

From here we can conclude that $c = \frac{1}{2}(c_1 + c_2) \approx \frac{1}{2}(v_a + v_b)$. The latter means that now c will approximately be $\frac{1}{2}(v_a + v_b)$ across different samples of the key vectors.