knn

December 18, 2025

```python
[23]: # This mounts your Google Drive to the Colab VM.
      from google.colab import drive
      drive.mount('/content/drive')

      # TODO: Enter the foldername in your Drive where you have saved the unzipped
      # assignment folder, e.g. 'icv83551/assignments/assignment1/'
      FOLDERNAME = 'icv83551/assignments/assignment1/'
      assert FOLDERNAME is not None, "[!] Enter the foldername."

      # Now that we've mounted your Drive, this ensures that
      # the Python interpreter of the Colab VM can load
      # python files from within it.
      import sys
      sys.path.append('/content/drive/My Drive/{}'.format(FOLDERNAME))

      # This downloads the CIFAR-10 dataset to your Drive
      # if it doesn't already exist.
      %cd /content/drive/My\ Drive/$FOLDERNAME/icv83551/datasets/
      !bash get_datasets.sh
      %cd /content/drive/My\ Drive/$FOLDERNAME
```

Drive already mounted at /content/drive; to attempt to forcibly remount, call
drive.mount("/content/drive", force_remount=True).
/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/datasets
/content/drive/My Drive/icv83551/assignments/assignment1

1 k-Nearest Neighbor (kNN) exercise

*Complete and hand in this completed worksheet (including its outputs and any supporting code outside of the worksheet) with your assignment submission. The kNN classifier consists of two stages:

- During training, the classifier takes the training data and simply remembers it
- During testing, kNN classifies every test image by comparing to all training images and transfering the labels of the k most similar training examples
- The value of k is cross-validated

In this exercise you will implement these steps and understand the basic Image Classification pipeline, cross-validation, and gain proficiency in writing efficient, vectorized code.

```
[24]:  # Run some setup code for this notebook.

       import random
       import numpy as np
       from icv83551.data_utils import load_CIFAR10
       import matplotlib.pyplot as plt

       # This is a bit of magic to make matplotlib figures appear inline in the␣
        ↪notebook
       # rather than in a new window.
       %matplotlib inline
       plt.rcParams['figure.figsize'] = (10.0, 8.0) # set default size of plots
       plt.rcParams['image.interpolation'] = 'nearest'
       plt.rcParams['image.cmap'] = 'gray'

       #%load_ext autoreload
       #%autoreload 2
```

```
[25]:  # Load the raw CIFAR-10 data.
       cifar10_dir = 'icv83551/datasets/cifar-10-batches-py'

       # Cleaning up variables to prevent loading data multiple times (which may cause␣
        ↪memory issue)
       try:
          del X_train, y_train
          del X_test, y_test
          print('Clear previously loaded data.')
       except:
          pass

       X_train, y_train, X_test, y_test = load_CIFAR10(cifar10_dir)

       # As a sanity check, we print out the size of the training and test data.
       print('Training data shape: ', X_train.shape)
       print('Training labels shape: ', y_train.shape)
       print('Test data shape: ', X_test.shape)
       print('Test labels shape: ', y_test.shape)
```

```
Clear previously loaded data.
Training data shape:  (50000, 32, 32, 3)
Training labels shape:  (50000,)
Test data shape:  (10000, 32, 32, 3)
Test labels shape:  (10000,)
```

```
[26]:  # Visualize some examples from the dataset.
       # We show a few examples of training images from each class.
```

```
classes = ['plane', 'car', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse',␣
 ↪'ship', 'truck']
num_classes = len(classes)
samples_per_class = 7
for y, cls in enumerate(classes):
    idxs = np.flatnonzero(y_train == y)
    idxs = np.random.choice(idxs, samples_per_class, replace=False)
    for i, idx in enumerate(idxs):
        plt_idx = i * num_classes + y + 1
        plt.subplot(samples_per_class, num_classes, plt_idx)
        plt.imshow(X_train[idx].astype('uint8'))
        plt.axis('off')
        if i == 0:
            plt.title(cls)
plt.show()
```
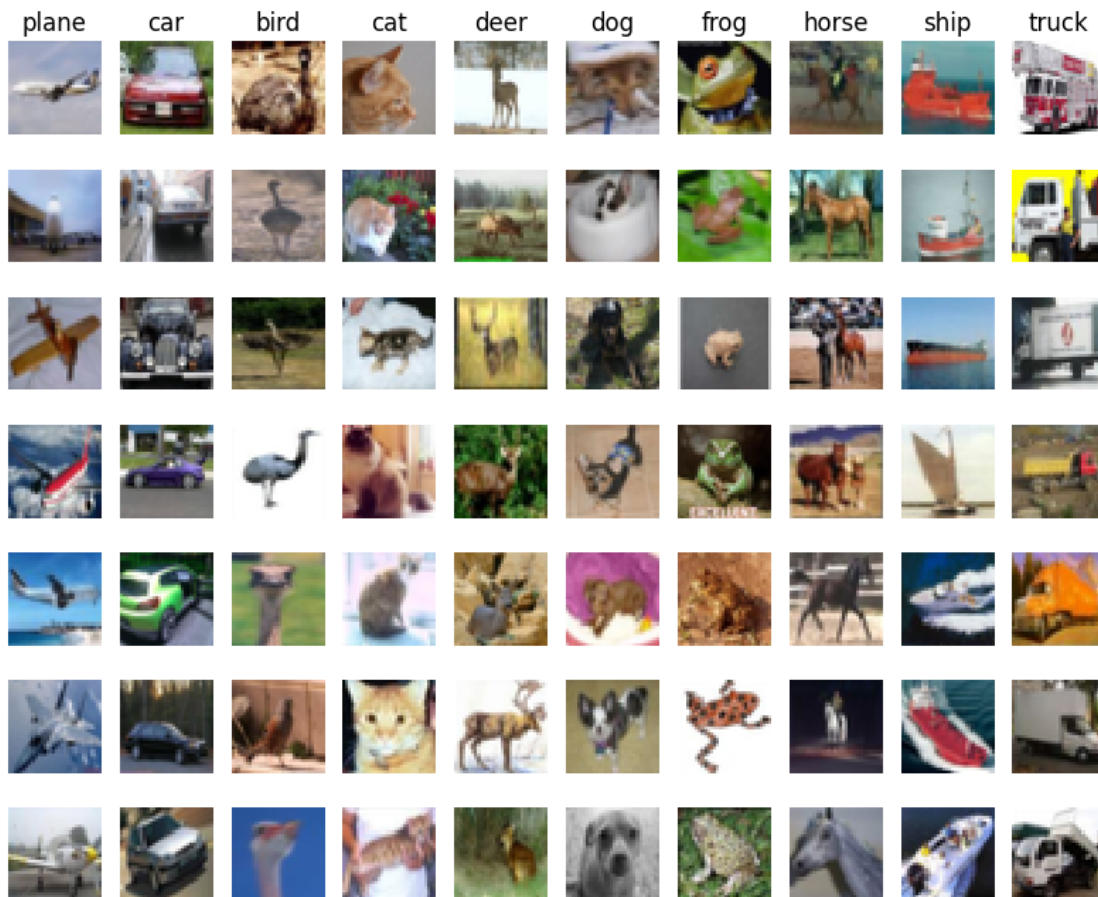


```
[27]: # Subsample the data for more efficient code execution in this exercise
      num_training = 5000
```

```
mask = list(range(num_training))
X_train = X_train[mask]
y_train = y_train[mask]

num_test = 500
mask = list(range(num_test))
X_test = X_test[mask]
y_test = y_test[mask]

# Reshape the image data into rows
X_train = np.reshape(X_train, (X_train.shape[0], -1))
X_test = np.reshape(X_test, (X_test.shape[0], -1))
print(X_train.shape, X_test.shape)
```

```
(5000, 3072) (500, 3072)
```

[28]:
```
from icv83551.classifiers import KNearestNeighbor

# Create a kNN classifier instance.
# Remember that training a kNN classifier is a noop:
# 'noop' stands for "no operation": in this context, training a kNN classifier
# does not perform any real learning step, it simply stores the labeled data
# so it can be used later to find nearest neighbors during prediction.

# the Classifier simply remembers the data and does no further processing
classifier = KNearestNeighbor()
classifier.train(X_train, y_train)
```

We would now like to classify the test data with the kNN classifier. Recall that we can break down this process into two steps:

1. First we must compute the distances between all test examples and all train examples.
2. Given these distances, for each test example we find the k nearest examples and have them vote for the label

Lets begin with computing the distance matrix between all training and test examples. For example, if there are **Ntr** training examples and **Nte** test examples, this stage should result in a **Nte x Ntr** matrix where each element (i,j) is the distance between the i-th test and j-th train example.

Note: For the three distance computations that we require you to implement in this notebook, you may not use the np.linalg.norm() function that numpy provides.

First, open `icv83551/classifiers/k_nearest_neighbor.py` and implement the function `compute_distances_two_loops` that uses a (very inefficient) double loop over all pairs of (test, train) examples and computes the distance matrix one element at a time.

[29]:
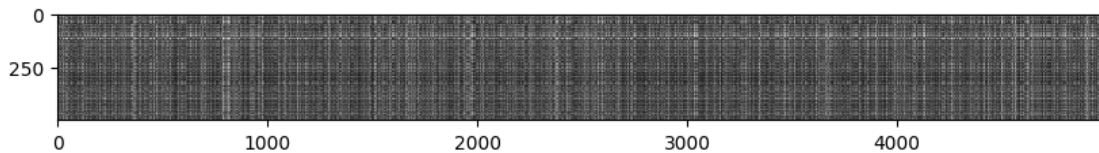```
Open icv83551/classifiers/k_nearest_neighbor.py and implement
compute_distances_two_loops.
```

```
# Test your implementation:
dists = classifier.compute_distances_two_loops(X_test)
print(dists.shape)
```

(500, 5000)

```
[30]: # We can visualize the distance matrix: each row is a single test example and
      # its distances to training examples
      plt.imshow(dists, interpolation='none')
      plt.show()
```


Inline Question 1

Notice the structured patterns in the distance matrix, where some rows or columns are visibly brighter. (Note that with the default color scheme black indicates low distances while white indicates high distances.)

- What in the data is the cause behind the distinctly bright rows?
- What causes the columns?

Your Answer : *the bright columns are when the test example is very distant (not similair) from the training examples to almost all training examples. this could happen if the test image has unique backroad color or distinct scale not seen in previous images, the row is the same, but the train is very distant (not similair) from the almost all training examples, this suggest that the training image is an outlier compared to the disterbution of the test set (examples)*

```
[31]: # Now implement the function predict_labels and run the code below:
      # We use k = 1 (which is Nearest Neighbor).
      y_test_pred = classifier.predict_labels(dists, k=1)

      # Compute and print the fraction of correctly predicted examples
      num_correct = np.sum(y_test_pred == y_test)
      accuracy = float(num_correct) / num_test
      print('Got %d / %d correct => accuracy: %f' % (num_correct, num_test, accuracy))
```

Got 137 / 500 correct => accuracy: 0.274000

You should expect to see approximately 27% accuracy. Now lets try out a larger k, say k = 5:

```
[32]: y_test_pred = classifier.predict_labels(dists, k=5)
      num_correct = np.sum(y_test_pred == y_test)
      accuracy = float(num_correct) / num_test
```

```
print('Got %d / %d correct => accuracy: %f' % (num_correct, num_test, accuracy))
```

`Got 139 / 500 correct => accuracy: 0.278000`

You should expect to see a slightly better performance than with `k = 1`.

Inline Question 2

We can also use other distance metrics such as L1 distance. For pixel values $p_{ij}^{(k)}$ at location (i, j) of some image I_k,

the mean μ across all pixels over all images is

$$\mu = \frac{1}{nhw} \sum_{k=1}^{n} \sum_{i=1}^{h} \sum_{j=1}^{w} p_{ij}^{(k)}$$

And the pixel-wise mean μ_{ij} across all images is

$$\mu_{ij} = \frac{1}{n} \sum_{k=1}^{n} p_{ij}^{(k)}.$$

The general standard deviation σ and pixel-wise standard deviation σ_{ij} is defined similarly.

Which of the following preprocessing steps will not change the performance of a Nearest Neighbor classifier that uses L1 distance? Select all that apply. To clarify, both training and test examples are preprocessed in the same way.

1. Subtracting the mean μ ($\tilde{p}_{ij}^{(k)} = p_{ij}^{(k)} - \mu$.)
2. Subtracting the per pixel mean μ_{ij} ($\tilde{p}_{ij}^{(k)} = p_{ij}^{(k)} - \mu_{ij}$.)
3. Subtracting the mean μ and dividing by the standard deviation σ.
4. Subtracting the pixel-wise mean μ_{ij} and dividing by the pixel-wise standard deviation σ_{ij}.
5. Rotating the coordinate axes of the data, which means rotating all the images by the same angle. Empty regions in the image caused by rotation are padded with a same pixel value and no interpolation is performed.

Your Answer : 1,2,3

Your Explanation : for 1&2 we have the invarince in subtraction with the same subtraction, meaning, for: 1. $|(p_{ij}^{(k)} - \mu) - (p_{ij}^{(l)} - \mu)| = |(p_{ij}^{(k)} - p_{ij}^{(l)}) + (-\mu + \mu)| = |p_{ij}^{(k)} - p_{ij}^{(l)}|$ 2. $|(p_{ij}^{(k)} - \mu_{ij}) - (p_{ij}^{(l)} - \mu_{ij})| = |(p_{ij}^{(k)} - p_{ij}^{(l)}) + (-\mu_{ij} + \mu_{ij})| = |(p_{ij}^{(k)} - p_{ij}^{(l)})|$

for 3 - we have a scaling by σ^{-1} and the invarient subtraction, but we can see that if point A was closer then point B, then scaling all points by σ^{-1} wont change the distance:

3. $|((p_{ij}^{(k)} - \mu)\sigma^{-1}) - ((p_{ij}^{(l)} - \mu)\sigma^{-1})| = |((p_{ij}^{(k)} - p_{ij}^{(l)}) + (-\mu + \mu))\sigma^{-1}| = |(p_{ij}^{(k)} - p_{ij}^{(l)})\sigma^{-1}|$

we can see that all the distances are scaled by sigma^{-1} meaning there place in the order does not change (uniform scaling across all pixels, which doesnt change there reletive distances)

for 4 - it is incorrect since each pixel variance is diffrent, meaning it scales each pixel diffrence differently. this means we "assign" a weight to every pixel baised on it variance. this ditorts the "shape" of the feature in a non uniform way (unlike 3), altering the differences, which in turn changes which neighbor is ranked as the closest.

6

for 5 - l1 is not a circle metric, meaning it does not hold a invariance for rotation (unlike L2 which is invariant - since all the point of a certain distance are a cirle). a great example i distance 1, the point (1,0) which is rotated by 45 degrees and gives us (2/sqrt2, 2/sqrt2) and the L1 distance is sqrt2.

```python
[33]: # Now lets speed up distance matrix computation by using partial vectorization
      # with one loop. Implement the function compute_distances_one_loop and run the
      # code below:
      dists_one = classifier.compute_distances_one_loop(X_test)

      # To ensure that our vectorized implementation is correct, we make sure that it
      # agrees with the naive implementation. There are many ways to decide whether
      # two matrices are similar; one of the simplest is the Frobenius norm. In case
      # you haven't seen it before, the Frobenius norm of two matrices is the square
      # root of the squared sum of differences of all elements; in other words,␣
      ↪reshape
      # the matrices into vectors and compute the Euclidean distance between them.
      difference = np.linalg.norm(dists - dists_one, ord='fro')
      print('One loop difference was: %f' % (difference, ))
      if difference < 0.001:
          print('Good! The distance matrices are the same')
      else:
          print('Uh-oh! The distance matrices are different')
```

```
One loop difference was: 0.000000
Good! The distance matrices are the same
```

```python
[34]: # Now implement the fully vectorized version inside compute_distances_no_loops
      # and run the code
      dists_two = classifier.compute_distances_no_loops(X_test)

      # check that the distance matrix agrees with the one we computed before:
      difference = np.linalg.norm(dists - dists_two, ord='fro')
      print('No loop difference was: %f' % (difference, ))
      if difference < 0.001:
          print('Good! The distance matrices are the same')
      else:
          print('Uh-oh! The distance matrices are different')
```

```
No loop difference was: 0.000000
Good! The distance matrices are the same
```

```python
[35]: # Let's compare how fast the implementations are
      def time_function(f, *args):
          """
          Call a function f with args and return the time (in seconds) that it took␣
      ↪to execute.
          """
```

```
    import time
    tic = time.time()
    f(*args)
    toc = time.time()
    return toc - tic


two_loop_time = time_function(classifier.compute_distances_two_loops, X_test)
print('Two loop version took %f seconds' % two_loop_time)


one_loop_time = time_function(classifier.compute_distances_one_loop, X_test)
print('One loop version took %f seconds' % one_loop_time)


no_loop_time = time_function(classifier.compute_distances_no_loops, X_test)
print('No loop version took %f seconds' % no_loop_time)


# You should see significantly faster performance with the fully vectorized␣
 ↪implementation!


# NOTE: depending on what machine you're using,
# you might not see a speedup when you go from two loops to one loop,
# and might even see a slow-down.
```

```
Two loop version took 38.102354 seconds
One loop version took 54.401847 seconds
No loop version took 1.134438 seconds
```

1.0.1 Cross-validation

We have implemented the k-Nearest Neighbor classifier but we set the value k = 5 arbitrarily. We will now determine the best value of this hyperparameter with cross-validation.

```
[36]: num_folds = 5
    k_choices = [1, 3, 5, 8, 10, 12, 15, 20, 50, 100]

    X_train_folds = []
    y_train_folds = []
    ################################################################################
    # TODO:                                                                        #
    # Split up the training data into folds. After splitting, X_train_folds and    #
    # y_train_folds should each be lists of length num_folds, where                #
    # y_train_folds[i] is the label vector for the points in X_train_folds[i].     #
    # Hint: Look up the numpy array_split function.                                 #
    ################################################################################
    X_folds = np.array_split(X_train, num_folds)
    Y_folds = np.array_split(y_train, num_folds)
    ################################################################################

    # A dictionary holding the accuracies for different values of k that we find
```

```python
# when running cross-validation. After running cross-validation,
# k_to_accuracies[k] should be a list of length num_folds giving the different
# accuracy values that we found when using that value of k.

k_to_accuracies = {}
################################################################################
# TODO:                                                                        #
# Perform k-fold cross validation to find the best value of k. For each        #
# possible value of k, run the k-nearest-neighbor algorithm num_folds times,   #
# where in each case you use all but one of the folds as training data and the #
# last fold as a validation set. Store the accuracies for all fold and all     #
# values of k in the k_to_accuracies dictionary.                               #
################################################################################
for k in k_choices:
  k_to_accuracies[k] = []
  for i in range(num_folds):
    X_val_fold = X_folds[i]
    Y_val_fold =Y_folds[i]

    X_train_curr = np.concatenate(X_folds[:i] + X_folds[i+1:])
    Y_train_curr = np.concatenate(Y_folds[:i] + Y_folds[i+1:])

    classifier.train(X_train_curr, Y_train_curr)
    dists = classifier.compute_distances_no_loops(X_val_fold)
    y_test_pred = classifier.predict_labels(dists, k)
    num_correct = np.sum(y_test_pred == Y_val_fold)
    accuracy = float(num_correct) / Y_val_fold.shape[0]
    k_to_accuracies[k].append(accuracy)

# Print out the computed accuracies
for k in sorted(k_to_accuracies):
    for accuracy in k_to_accuracies[k]:
        print('k = %d, accuracy = %f' % (k, accuracy))
```

```
k = 1, accuracy = 0.263000
k = 1, accuracy = 0.257000
k = 1, accuracy = 0.264000
k = 1, accuracy = 0.278000
k = 1, accuracy = 0.266000
k = 3, accuracy = 0.239000
k = 3, accuracy = 0.249000
k = 3, accuracy = 0.240000
k = 3, accuracy = 0.266000
k = 3, accuracy = 0.254000
k = 5, accuracy = 0.248000
k = 5, accuracy = 0.266000
k = 5, accuracy = 0.280000
```

```
k = 5, accuracy = 0.292000
k = 5, accuracy = 0.280000
k = 8, accuracy = 0.262000
k = 8, accuracy = 0.282000
k = 8, accuracy = 0.273000
k = 8, accuracy = 0.290000
k = 8, accuracy = 0.273000
k = 10, accuracy = 0.265000
k = 10, accuracy = 0.296000
k = 10, accuracy = 0.276000
k = 10, accuracy = 0.284000
k = 10, accuracy = 0.280000
k = 12, accuracy = 0.260000
k = 12, accuracy = 0.295000
k = 12, accuracy = 0.279000
k = 12, accuracy = 0.283000
k = 12, accuracy = 0.280000
k = 15, accuracy = 0.252000
k = 15, accuracy = 0.289000
k = 15, accuracy = 0.278000
k = 15, accuracy = 0.282000
k = 15, accuracy = 0.274000
k = 20, accuracy = 0.270000
k = 20, accuracy = 0.279000
k = 20, accuracy = 0.279000
k = 20, accuracy = 0.282000
k = 20, accuracy = 0.285000
k = 50, accuracy = 0.271000
k = 50, accuracy = 0.288000
k = 50, accuracy = 0.278000
k = 50, accuracy = 0.269000
k = 50, accuracy = 0.266000
k = 100, accuracy = 0.256000
k = 100, accuracy = 0.270000
k = 100, accuracy = 0.263000
k = 100, accuracy = 0.256000
k = 100, accuracy = 0.263000
```

```python
[37]: # plot the raw observations
      for k in k_choices:
          accuracies = k_to_accuracies[k]
          plt.scatter([k] * len(accuracies), accuracies)

      # plot the trend line with error bars that correspond to standard deviation
      accuracies_mean = np.array([np.mean(v) for k,v in sorted(k_to_accuracies.
        ↪items())])
```

```
accuracies_std = np.array([np.std(v) for k,v in sorted(k_to_accuracies.
 ↪items())])
plt.errorbar(k_choices, accuracies_mean, yerr=accuracies_std)
plt.title('Cross-validation on k')
plt.xlabel('k')
plt.ylabel('Cross-validation accuracy')
plt.grid()
plt.legend()
plt.show()
```

/tmp/ipython-input-2604147651.py:14: UserWarning: No artists with labels found
to put in legend. Note that artists whose label start with an underscore are
ignored when legend() is called with no argument.
  plt.legend()

Cross-validation on k
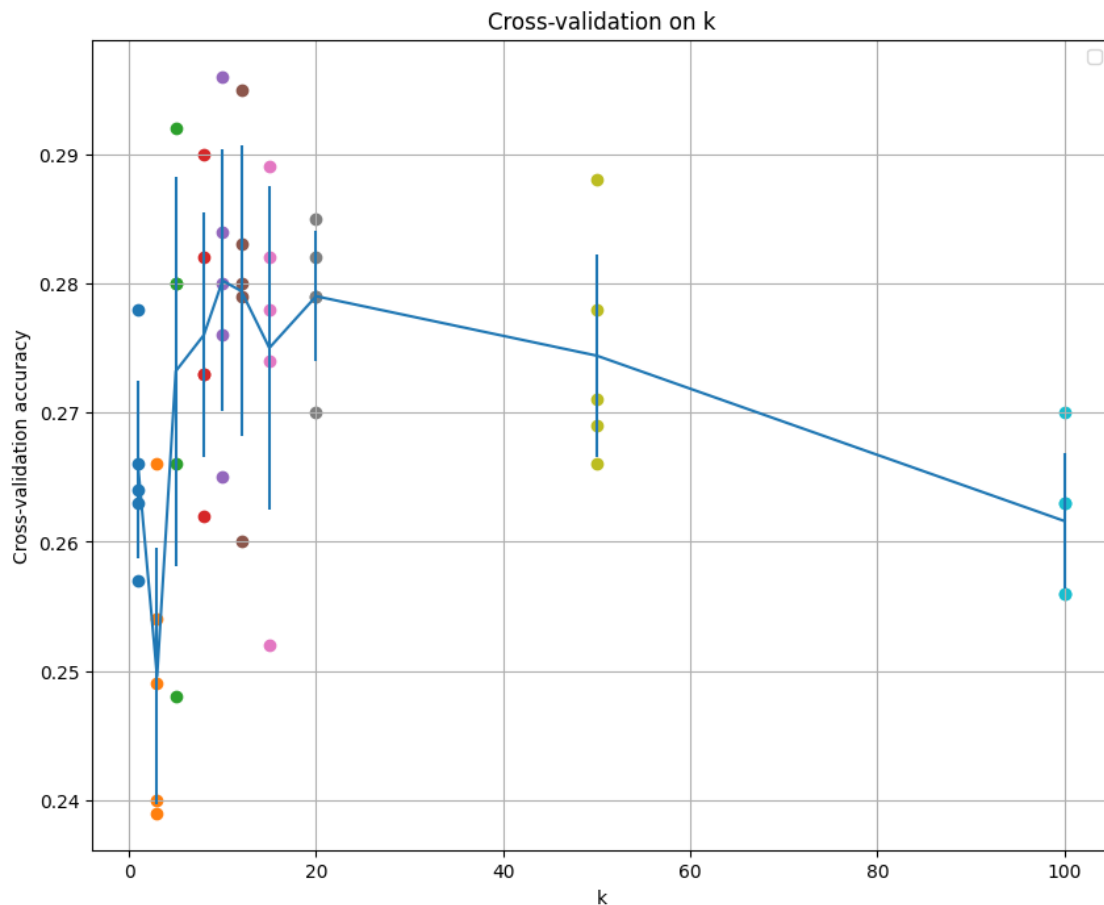
[38]: ```
Based on the cross-validation results above, choose the best value for k,
retrain the classifier using all the training data, and test it on the test
data. You should be able to get above 28% accuracy on the test data.
```

11

```
best_k = 10

classifier = KNearestNeighbor()
classifier.train(X_train, y_train)
y_test_pred = classifier.predict(X_test, k=best_k)

# Compute and display the accuracy
num_correct = np.sum(y_test_pred == y_test)
accuracy = float(num_correct) / num_test
print('Got %d / %d correct => accuracy: %f' % (num_correct, num_test, accuracy))
```

Got 141 / 500 correct => accuracy: 0.282000

Inline Question 3

Which of the following statements about k-Nearest Neighbor (k-NN) are true in a classification setting, and for all k? Select all that apply. 1. The decision boundary of the k-NN classifier is linear. 2. The training error of a 1-NN will always be lower than or equal to that of 5-NN. 3. The test error of a 1-NN will always be lower than that of a 5-NN. 4. The time needed to classify a test example with the k-NN classifier grows with the size of the training set. 5. None of the above.

Your Answer : number 4 is correct

Your Explanation : 1. False - on the base of what we learnt - the decision boundry is extremly non linear, as it adapts to the local geomitry of the training data, forming non linaer polygonal shapes. we saw this in the call where we spoke of overfitting. 2. True - in 1-NN the closest neighbor to each point is itself, making during training it will always predict the correct label, meaning the training loss error of 0. since we know that the error is always positive or 0, 5-NN will always be equal or more then 1-NN. 3. False - we saw in this project that k=5 had higher accuracy - therefore lower error, this comes from the fact that 1-NN is extremly sensetive to outliers, making it very prone to overfitting, effectively memorizing the training data. 4. True - as the size grows, the amount of distances needed to calculate grows, taking more time (more of the same operations - take more time). the calculation of the distances is done in O(N), meaning it is linearly connected to the size of the test set, therefore takes longer as the test set grows. 5. False - 2 &4 are true, making this false.

12

softmax

December 18, 2025

```python
[61]: # This mounts your Google Drive to the Colab VM.
      from google.colab import drive
      drive.mount('/content/drive')

      # TODO: Enter the foldername in your Drive where you have saved the unzipped
      # assignment folder, e.g. 'icv83551/assignments/assignment1/'
      FOLDERNAME = 'icv83551/assignments/assignment1/'
      assert FOLDERNAME is not None, "[!] Enter the foldername."

      # Now that we've mounted your Drive, this ensures that
      # the Python interpreter of the Colab VM can load
      # python files from within it.
      import sys
      sys.path.append('/content/drive/My Drive/{}'.format(FOLDERNAME))

      # This downloads the CIFAR-10 dataset to your Drive
      # if it doesn't already exist.
      %cd /content/drive/My\ Drive/$FOLDERNAME/icv83551/datasets/
      !bash get_datasets.sh
      %cd /content/drive/My\ Drive/$FOLDERNAME
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call
drive.mount("/content/drive", force_remount=True).
/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/datasets
/content/drive/My Drive/icv83551/assignments/assignment1
```

1 Softmax Classifier exercise

*Complete and hand in this completed worksheet (including its outputs and any supporting code outside of the worksheet) with your assignment submission.

In this exercise you will:

- Implement a fully-vectorized **loss function** for the Softmax classifier.
- Implement the fully-vectorized expression for its **analytic gradient**
- **Check your implementation** using numerical gradient
- Use a validation set to **tune the learning rate and regularization** strength
- **Optimize** the loss function with **SGD**
- **Visualize** the final learned weights

```python
[62]: # Run some setup code for this notebook.
      import random
      import numpy as np
      from icv83551.data_utils import load_CIFAR10
      import matplotlib.pyplot as plt

      # This is a bit of magic to make matplotlib figures appear inline in the
      # notebook rather than in a new window.
      %matplotlib inline
      plt.rcParams['figure.figsize'] = (10.0, 8.0) # set default size of plots
      plt.rcParams['image.interpolation'] = 'nearest'
      plt.rcParams['image.cmap'] = 'gray'

      #%load_ext autoreload
      #%autoreload 2
```

1.1 CIFAR-10 Data Loading and Preprocessing

```python
[63]: # Load the raw CIFAR-10 data.
      cifar10_dir = 'icv83551/datasets/cifar-10-batches-py'

      # Cleaning up variables to prevent loading data multiple times (which may cause␣
       ↪memory issue)
      try:
         del X_train, y_train
         del X_test, y_test
         print('Clear previously loaded data.')
      except:
         pass

      X_train, y_train, X_test, y_test = load_CIFAR10(cifar10_dir)

      # As a sanity check, we print out the size of the training and test data.
      print('Training data shape: ', X_train.shape)
      print('Training labels shape: ', y_train.shape)
      print('Test data shape: ', X_test.shape)
      print('Test labels shape: ', y_test.shape)
```

```
Clear previously loaded data.
Training data shape:  (50000, 32, 32, 3)
Training labels shape:  (50000,)
Test data shape:  (10000, 32, 32, 3)
Test labels shape:  (10000,)
```

```python
[64]: # Visualize some examples from the dataset.
      # We show a few examples of training images from each class.
```

```python
classes = ['plane', 'car', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse',
    'ship', 'truck']
num_classes = len(classes)
samples_per_class = 7
for y, cls in enumerate(classes):
    idxs = np.flatnonzero(y_train == y)
    idxs = np.random.choice(idxs, samples_per_class, replace=False)
    for i, idx in enumerate(idxs):
        plt_idx = i * num_classes + y + 1
        plt.subplot(samples_per_class, num_classes, plt_idx)
        plt.imshow(X_train[idx].astype('uint8'))
        plt.axis('off')
        if i == 0:
            plt.title(cls)
plt.show()
```



```python
[65]:  # Split the data into train, val, and test sets. In addition we will
       # create a small development set as a subset of the training data;
```

```python
# we can use this for development so our code runs faster.
num_training = 49000
num_validation = 1000
num_test = 1000
num_dev = 500

# Our validation set will be num_validation points from the original
# training set.
mask = range(num_training, num_training + num_validation)
X_val = X_train[mask]
y_val = y_train[mask]

# Our training set will be the first num_train points from the original
# training set.
mask = range(num_training)
X_train = X_train[mask]
y_train = y_train[mask]

# We will also make a development set, which is a small subset of
# the training set.
mask = np.random.choice(num_training, num_dev, replace=False)
X_dev = X_train[mask]
y_dev = y_train[mask]

# We use the first num_test points of the original test set as our
# test set.
mask = range(num_test)
X_test = X_test[mask]
y_test = y_test[mask]

print('Train data shape: ', X_train.shape)
print('Train labels shape: ', y_train.shape)
print('Validation data shape: ', X_val.shape)
print('Validation labels shape: ', y_val.shape)
print('Test data shape: ', X_test.shape)
print('Test labels shape: ', y_test.shape)
```

```
Train data shape:  (49000, 32, 32, 3)
Train labels shape:  (49000,)
Validation data shape:  (1000, 32, 32, 3)
Validation labels shape:  (1000,)
Test data shape:  (1000, 32, 32, 3)
Test labels shape:  (1000,)
```

```python
[66]: # Preprocessing: reshape the image data into rows
X_train = np.reshape(X_train, (X_train.shape[0], -1))
X_val = np.reshape(X_val, (X_val.shape[0], -1))
```

```
X_test = np.reshape(X_test, (X_test.shape[0], -1))
X_dev = np.reshape(X_dev, (X_dev.shape[0], -1))

# As a sanity check, print out the shapes of the data
print('Training data shape: ', X_train.shape)
print('Validation data shape: ', X_val.shape)
print('Test data shape: ', X_test.shape)
print('dev data shape: ', X_dev.shape)
```

```
Training data shape:  (49000, 3072)
Validation data shape:  (1000, 3072)
Test data shape:  (1000, 3072)
dev data shape:  (500, 3072)
```

[67]:
```
# Preprocessing: subtract the mean image
# first: compute the image mean based on the training data
mean_image = np.mean(X_train, axis=0)
print(mean_image[:10]) # print a few of the elements
plt.figure(figsize=(4,4))
plt.imshow(mean_image.reshape((32,32,3)).astype('uint8')) # visualize the mean
 ↪image
plt.show()

# second: subtract the mean image from train and test data
X_train -= mean_image
X_val -= mean_image
X_test -= mean_image
X_dev -= mean_image

# third: append the bias dimension of ones (i.e. bias trick) so that our
 ↪classifier
# only has to worry about optimizing a single weight matrix W.
X_train = np.hstack([X_train, np.ones((X_train.shape[0], 1))])
X_val = np.hstack([X_val, np.ones((X_val.shape[0], 1))])
X_test = np.hstack([X_test, np.ones((X_test.shape[0], 1))])
X_dev = np.hstack([X_dev, np.ones((X_dev.shape[0], 1))])

print(X_train.shape, X_val.shape, X_test.shape, X_dev.shape)
```

```
[130.64189796 135.98173469 132.47391837 130.05569388 135.34804082
 131.75402041 130.96055102 136.14328571 132.47636735 131.48467347]
```
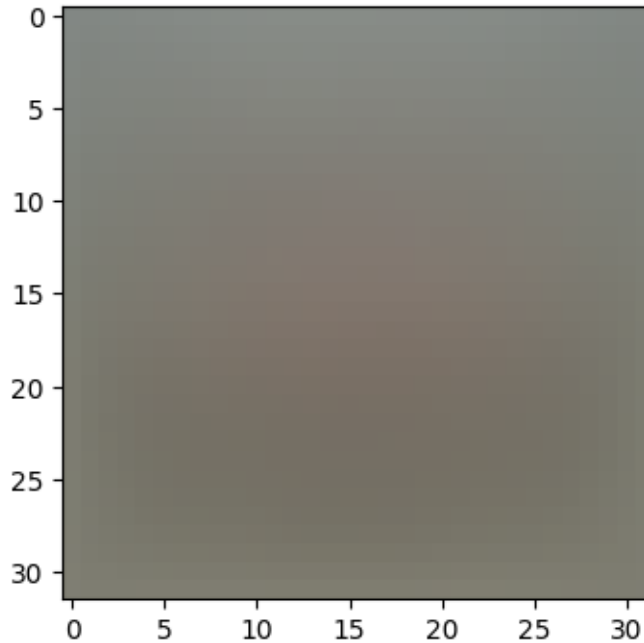
(49000, 3073) (1000, 3073) (1000, 3073) (500, 3073)

1.2 Softmax Classifier

Your code for this section will all be written inside `icv83551/classifiers/softmax.py`.

As you can see, we have prefilled the function `softmax_loss_naive` which uses for loops to evaluate the softmax loss function.

```
[68]: # Evaluate the naive implementation of the loss we provided for you:
      from icv83551.classifiers.softmax import softmax_loss_naive
      import time

      # generate a random Softmax classifier weight matrix of small numbers
      W = np.random.randn(3073, 10) * 0.0001

      loss, grad = softmax_loss_naive(W, X_dev, y_dev, 0.000005)
      print('loss: %f' % (loss, ))

      # As a rough sanity check, our loss should be something close to -log(0.1).
      print('loss: %f' % loss)
      print('sanity check: %f' % (-np.log(0.1)))
```

```
loss: 2.361920
loss: 2.361920
sanity check: 2.302585
```

6

Inline Question 1

Why do we expect our loss to be close to -log(0.1)? Explain briefly.**

Your Answer : we initialize W randomly close to zero, meaning: s=Wx will be approximatly 0 for all x, and we know that the softmax function is $e^{x_i}/sum(e^{x_j}) =_{around} 1/10$, and using the cross entrophy loss we get that it is equal to around -ln(0.1).

The `grad` returned from the function above is right now all zero. Derive and implement the gradient for the softmax loss function and implement it inline inside the function `softmax_loss_naive`. You will find it helpful to interleave your new code inside the existing function.

To check that you have correctly implemented the gradient, you can numerically estimate the gradient of the loss function and compare the numeric estimate to the gradient that you computed. We have provided code that does this for you:

```
[69]:  # Once you've implemented the gradient, recompute it with the code below
       # and gradient check it with the function we provided for you

       # Compute the loss and its gradient at W.
       loss, grad = softmax_loss_naive(W, X_dev, y_dev, 0.0)

       # Numerically compute the gradient along several randomly chosen dimensions, and
       # compare them with your analytically computed gradient. The numbers should
         ↪match
       # almost exactly along all dimensions.
       from icv83551.gradient_check import grad_check_sparse
       f = lambda w: softmax_loss_naive(w, X_dev, y_dev, 0.0)[0]
       grad_numerical = grad_check_sparse(f, W, grad)

       # do the gradient check once again with regularization turned on
       # you didn't forget the regularization gradient did you?
       loss, grad = softmax_loss_naive(W, X_dev, y_dev, 5e1)
       f = lambda w: softmax_loss_naive(w, X_dev, y_dev, 5e1)[0]
       grad_numerical = grad_check_sparse(f, W, grad)
```

```
numerical: -0.680736 analytic: -0.680736, relative error: 2.549904e-08
numerical: -3.535919 analytic: -3.535919, relative error: 3.190834e-12
numerical: -0.265233 analytic: -0.265233, relative error: 6.302939e-10
numerical: -2.841765 analytic: -2.841765, relative error: 1.360027e-09
numerical: -0.027354 analytic: -0.027354, relative error: 5.772209e-07
numerical: -0.213063 analytic: -0.213063, relative error: 1.494791e-07
numerical: 2.169244 analytic: 2.169244, relative error: 2.809594e-08
numerical: -0.262223 analytic: -0.262223, relative error: 3.267684e-08
numerical: -1.061632 analytic: -1.061632, relative error: 6.662123e-09
numerical: 3.022438 analytic: 3.022438, relative error: 1.388523e-08
numerical: -0.496273 analytic: -0.496273, relative error: 8.420391e-08
numerical: 0.527239 analytic: 0.527239, relative error: 5.679826e-08
numerical: -2.328130 analytic: -2.328131, relative error: 3.023132e-08
numerical: 2.613398 analytic: 2.613398, relative error: 8.010130e-09
```

```
numerical: 0.860365 analytic: 0.860365, relative error: 4.365024e-08
numerical: 2.067532 analytic: 2.067532, relative error: 9.046411e-09
numerical: 0.831973 analytic: 0.831973, relative error: 3.915724e-08
numerical: 0.106859 analytic: 0.106859, relative error: 2.609832e-07
numerical: -2.129331 analytic: -2.129331, relative error: 2.697380e-08
numerical: -2.646475 analytic: -2.646476, relative error: 2.011433e-08
```

[69]:

Inline Question 2

Although gradcheck is reliable softmax loss, it is possible that for SVM loss, once in a while, a dimension in the gradcheck will not match exactly. What could such a discrepancy be caused by? Is it a reason for concern? What is a simple example in one dimension where a svm loss gradient check could fail? How would change the margin affect of the frequency of this happening?

Note that SVM loss for a sample (x_i, y_i) is defined as:

$$L_i = \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + \Delta)$$

Where j iterates over all classes except the correct class y_i and s_j denotes the classifier score for j^{th} class. Δ is a scalar margin. *Hint: the SVM loss function is not strictly speaking differentiable.*

Your Answer : in the analytical grad we pick a side in a arbetrery manner, therefor we get a binary choice between 1 and 0 (because: max(x,0)). when we solve numerically we use a small h and use the defenition of difrentation, meaning we will get some number close to 0.5, and cleary these dont match. this will not change the convergence of the function. Changing the margin of Δ will effect the frequincy. in the beggeining we randomly initailize the weights close to zero, meaning most of our numbers are close to zero, therefore if - $\Delta =_{closeto} 0$ then more of the numbers will be at the non diffrentable (0,0) point. if we put $\Delta >> 0$ then the likelyhood of getting there are much smaller. simple example:

$x = 10^{-6}, h = 10^{-4}$

the analytical graindent is 1 because of: $x > 0$

but the numerical graident is $f(x+h)/2h = (10^{-4} + 10^{-6})/2 * 10^{-4} = (1 + 10^{-2})/2$

and cleary they dont equal each other.

[70]:
```python
# Next implement the function softmax_loss_vectorized; for now only compute the
  ↪loss;
# we will implement the gradient in a moment.
tic = time.time()
loss_naive, grad_naive = softmax_loss_naive(W, X_dev, y_dev, 0.000005)
toc = time.time()
print('Naive loss: %e computed in %fs' % (loss_naive, toc - tic))

from icv83551.classifiers.softmax import softmax_loss_vectorized
tic = time.time()
loss_vectorized, _ = softmax_loss_vectorized(W, X_dev, y_dev, 0.000005)
```

```
toc = time.time()
print('Vectorized loss: %e computed in %fs' % (loss_vectorized, toc - tic))

# The losses should match but your vectorized implementation should be much
 ↪faster.
print('difference: %f' % (loss_naive - loss_vectorized))
```

Naive loss: 2.361920e+00 computed in 0.082124s
Vectorized loss: 2.361920e+00 computed in 0.012699s
difference: 0.000000

[71]:
```
# Complete the implementation of softmax_loss_vectorized, and compute the
 ↪gradient
# of the loss function in a vectorized way.

# The naive implementation and the vectorized implementation should match, but
# the vectorized version should still be much faster.
tic = time.time()
_, grad_naive = softmax_loss_naive(W, X_dev, y_dev, 0.000005)
toc = time.time()
print('Naive loss and gradient: computed in %fs' % (toc - tic))

tic = time.time()
_, grad_vectorized = softmax_loss_vectorized(W, X_dev, y_dev, 0.000005)
toc = time.time()
print('Vectorized loss and gradient: computed in %fs' % (toc - tic))

# The loss is a single number, so it is easy to compare the values computed
# by the two implementations. The gradient on the other hand is a matrix, so
# we use the Frobenius norm to compare them.
difference = np.linalg.norm(grad_naive - grad_vectorized, ord='fro')
print('difference: %f' % difference)
```

Naive loss and gradient: computed in 0.107950s
Vectorized loss and gradient: computed in 0.012725s
difference: 0.000000

1.2.1 Stochastic Gradient Descent

We now have vectorized and efficient expressions for the loss, the gradient and our gradient matches the numerical gradient. We are therefore ready to do SGD to minimize the loss. Your code for this part will be written inside icv83551/classifiers/linear_classifier.py.

[72]:
```
# In the file linear_classifier.py, implement SGD in the function
# LinearClassifier.train() and then run it with the code below.
from icv83551.classifiers import Softmax
softmax = Softmax()
tic = time.time()
```

```
loss_hist = softmax.train(X_train, y_train, learning_rate=1e-7, reg=2.5e4,
                          num_iters=1500, verbose=True)
toc = time.time()
print('That took %fs' % (toc - tic))
```

```
iteration 0 / 1500: loss 776.421473
iteration 100 / 1500: loss 284.574072
iteration 200 / 1500: loss 105.532048
iteration 300 / 1500: loss 39.867407
iteration 400 / 1500: loss 15.922339
iteration 500 / 1500: loss 7.164918
iteration 600 / 1500: loss 3.906926
iteration 700 / 1500: loss 2.794879
iteration 800 / 1500: loss 2.234702
iteration 900 / 1500: loss 2.195704
iteration 1000 / 1500: loss 2.096212
iteration 1100 / 1500: loss 2.048925
iteration 1200 / 1500: loss 2.058213
iteration 1300 / 1500: loss 2.068321
iteration 1400 / 1500: loss 2.097209
That took 11.790718s
```

```
[73]:  # A useful debugging strategy is to plot the loss as a function of
       # iteration number:
       plt.plot(loss_hist)
       plt.xlabel('Iteration number')
       plt.ylabel('Loss value')
       plt.grid()
       plt.show()
```
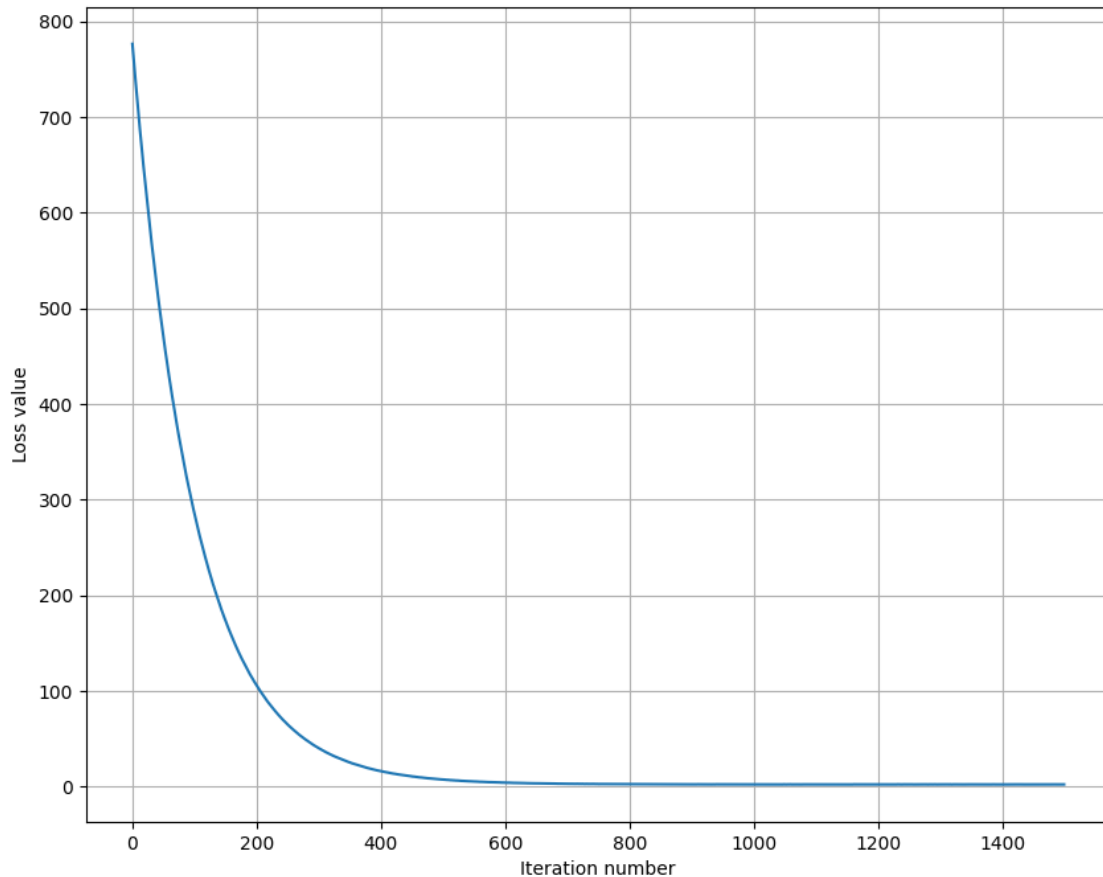
[74]:
```python
# Write the LinearClassifier.predict function and evaluate the performance on
# both the training and validation set
# You should get validation accuracy of about 0.34 (> 0.33).
y_train_pred = softmax.predict(X_train)
print('training accuracy: %f' % (np.mean(y_train == y_train_pred), ))
y_val_pred = softmax.predict(X_val)
print('validation accuracy: %f' % (np.mean(y_val == y_val_pred), ))
```

```
training accuracy: 0.319531
validation accuracy: 0.335000
```

[75]:
```python
# Save the trained model for autograder.
softmax.save("softmax.npy")
```

```
softmax.npy saved.
```

[76]:
```python
# Use the validation set to tune hyperparameters (regularization strength and
# learning rate). You should experiment with different ranges for the learning
# rates and regularization strengths; if you are careful you should be able to
```

11

```python
# get a classification accuracy of about 0.365 (> 0.36) on the validation set.

# Note: you may see runtime/overflow warnings during hyper-parameter search.
# This may be caused by extreme values, and is not a bug.

# results is dictionary mapping tuples of the form
# (learning_rate, regularization_strength) to tuples of the form
# (training_accuracy, validation_accuracy). The accuracy is simply the fraction
# of data points that are correctly classified.
results = {}
best_val = -1   # The highest validation accuracy that we have seen so far.
best_softmax = None # The Softmax object that achieved the highest validation
 ↪rate.

################################################################################
# TODO:                                                                        #
# Write code that chooses the best hyperparameters by tuning on the validation #
# set. For each combination of hyperparameters, train a Softmax on the.        #
# training set, compute its accuracy on the training and validation sets, and  #
# store these numbers in the results dictionary. In addition, store the best   #
# validation accuracy in best_val and the Softmax object that achieves this.   #
# accuracy in best_softmax.                                                     #
#                                                                              #
# Hint: You should use a small value for num_iters as you develop your         #
# validation code so that the classifiers don't take much time to train; once  #
# you are confident that your validation code works, you should rerun the      #
# code with a larger value for num_iters.                                      #
################################################################################

# Provided as a reference. You may or may not want to change these
 ↪hyperparameters
learning_rates = [1e-9,1e-8,1e-7, 1e-6,1e-5]
regularization_strengths = [5e4,2.5e4, 1e4, 0.5e4, 0.25e4]
for lr in learning_rates:
    for reg in regularization_strengths:
        softmax = Softmax()
        loss_hist = softmax.train(X_train, y_train, learning_rate=lr, reg=reg,
                    num_iters=1500, verbose=True)
        y_train_pred, y_val_pred = softmax.predict(X_train), softmax.
 ↪predict(X_val)
        (acc_train,acc_val) = (np.mean(y_train == y_train_pred), np.mean(y_val
 ↪== y_val_pred))
        results[(lr, reg)] = (acc_train,acc_val)
        if best_val < acc_val:
            best_val = acc_val
            best_softmax = softmax
```

```python
# Print out results.
for lr, reg in sorted(results):
    train_accuracy, val_accuracy = results[(lr, reg)]
    print('lr %e reg %e train accuracy: %f val accuracy: %f' % (
                lr, reg, train_accuracy, val_accuracy))

print('best validation accuracy achieved during cross-validation: %f' %␣
 ↪best_val)
```

```
iteration 0 / 1500: loss 1540.663162
iteration 100 / 1500: loss 1510.256832
iteration 200 / 1500: loss 1479.992512
iteration 300 / 1500: loss 1450.941127
iteration 400 / 1500: loss 1422.484423
iteration 500 / 1500: loss 1393.998721
iteration 600 / 1500: loss 1366.313848
iteration 700 / 1500: loss 1339.594992
iteration 800 / 1500: loss 1312.774899
iteration 900 / 1500: loss 1287.246904
iteration 1000 / 1500: loss 1261.176615
iteration 1100 / 1500: loss 1236.276775
iteration 1200 / 1500: loss 1211.875193
iteration 1300 / 1500: loss 1188.017879
iteration 1400 / 1500: loss 1163.960024
iteration 0 / 1500: loss 777.395207
iteration 100 / 1500: loss 769.860626
iteration 200 / 1500: loss 762.402073
iteration 300 / 1500: loss 754.514476
iteration 400 / 1500: loss 747.437894
iteration 500 / 1500: loss 739.857651
iteration 600 / 1500: loss 732.307545
iteration 700 / 1500: loss 724.907668
iteration 800 / 1500: loss 717.554778
iteration 900 / 1500: loss 710.712680
iteration 1000 / 1500: loss 703.555489
iteration 1100 / 1500: loss 696.367559
iteration 1200 / 1500: loss 689.395492
iteration 1300 / 1500: loss 682.829485
iteration 1400 / 1500: loss 675.830008
iteration 0 / 1500: loss 311.728912
iteration 100 / 1500: loss 310.752794
iteration 200 / 1500: loss 309.106003
iteration 300 / 1500: loss 308.158998
iteration 400 / 1500: loss 306.257923
iteration 500 / 1500: loss 305.294076
```

```
iteration 600 / 1500: loss 304.125644
iteration 700 / 1500: loss 302.945689
iteration 800 / 1500: loss 301.488153
iteration 900 / 1500: loss 300.437125
iteration 1000 / 1500: loss 299.748463
iteration 1100 / 1500: loss 298.409962
iteration 1200 / 1500: loss 296.738871
iteration 1300 / 1500: loss 295.650164
iteration 1400 / 1500: loss 293.737478
iteration 0 / 1500: loss 159.686263
iteration 100 / 1500: loss 159.795663
iteration 200 / 1500: loss 158.813744
iteration 300 / 1500: loss 158.619512
iteration 400 / 1500: loss 158.516330
iteration 500 / 1500: loss 158.050542
iteration 600 / 1500: loss 157.563040
iteration 700 / 1500: loss 157.507078
iteration 800 / 1500: loss 157.107064
iteration 900 / 1500: loss 157.109420
iteration 1000 / 1500: loss 156.147268
iteration 1100 / 1500: loss 156.114693
iteration 1200 / 1500: loss 155.472256
iteration 1300 / 1500: loss 155.551953
iteration 1400 / 1500: loss 154.904933
iteration 0 / 1500: loss 82.003256
iteration 100 / 1500: loss 82.036153
iteration 200 / 1500: loss 81.760774
iteration 300 / 1500: loss 80.987390
iteration 400 / 1500: loss 81.832335
iteration 500 / 1500: loss 81.743629
iteration 600 / 1500: loss 81.664132
iteration 700 / 1500: loss 81.368101
iteration 800 / 1500: loss 80.924836
iteration 900 / 1500: loss 81.079635
iteration 1000 / 1500: loss 80.884670
iteration 1100 / 1500: loss 80.892921
iteration 1200 / 1500: loss 81.380279
iteration 1300 / 1500: loss 80.625464
iteration 1400 / 1500: loss 80.777188
iteration 0 / 1500: loss 1543.410581
iteration 100 / 1500: loss 1263.380908
iteration 200 / 1500: loss 1033.888316
iteration 300 / 1500: loss 846.521818
iteration 400 / 1500: loss 693.088242
iteration 500 / 1500: loss 567.738597
iteration 600 / 1500: loss 464.977537
iteration 700 / 1500: loss 380.882504
iteration 800 / 1500: loss 312.167914
```

```
iteration 900 / 1500: loss 255.802500
iteration 1000 / 1500: loss 209.778283
iteration 1100 / 1500: loss 172.125343
iteration 1200 / 1500: loss 141.114771
iteration 1300 / 1500: loss 115.941665
iteration 1400 / 1500: loss 95.199469
iteration 0 / 1500: loss 781.674802
iteration 100 / 1500: loss 706.743745
iteration 200 / 1500: loss 639.548631
iteration 300 / 1500: loss 578.461116
iteration 400 / 1500: loss 523.298767
iteration 500 / 1500: loss 473.293906
iteration 600 / 1500: loss 428.330159
iteration 700 / 1500: loss 387.788020
iteration 800 / 1500: loss 350.777409
iteration 900 / 1500: loss 317.683985
iteration 1000 / 1500: loss 287.437169
iteration 1100 / 1500: loss 260.138975
iteration 1200 / 1500: loss 235.524752
iteration 1300 / 1500: loss 213.188949
iteration 1400 / 1500: loss 193.125802
iteration 0 / 1500: loss 313.963220
iteration 100 / 1500: loss 301.219245
iteration 200 / 1500: loss 289.132110
iteration 300 / 1500: loss 278.095054
iteration 400 / 1500: loss 266.999176
iteration 500 / 1500: loss 256.562371
iteration 600 / 1500: loss 246.496899
iteration 700 / 1500: loss 236.614500
iteration 800 / 1500: loss 227.289889
iteration 900 / 1500: loss 218.269873
iteration 1000 / 1500: loss 209.983985
iteration 1100 / 1500: loss 202.054141
iteration 1200 / 1500: loss 193.662361
iteration 1300 / 1500: loss 186.370993
iteration 1400 / 1500: loss 179.019361
iteration 0 / 1500: loss 158.327360
iteration 100 / 1500: loss 155.226070
iteration 200 / 1500: loss 152.129041
iteration 300 / 1500: loss 148.964243
iteration 400 / 1500: loss 144.988495
iteration 500 / 1500: loss 142.652710
iteration 600 / 1500: loss 139.408573
iteration 700 / 1500: loss 136.506207
iteration 800 / 1500: loss 134.031723
iteration 900 / 1500: loss 131.232253
iteration 1000 / 1500: loss 128.783028
iteration 1100 / 1500: loss 125.865242
```

```
iteration 1200 / 1500: loss 124.033313
iteration 1300 / 1500: loss 120.783011
iteration 1400 / 1500: loss 118.614823
iteration 0 / 1500: loss 82.034087
iteration 100 / 1500: loss 81.018264
iteration 200 / 1500: loss 80.233844
iteration 300 / 1500: loss 79.369062
iteration 400 / 1500: loss 77.806934
iteration 500 / 1500: loss 77.207174
iteration 600 / 1500: loss 76.310035
iteration 700 / 1500: loss 75.698121
iteration 800 / 1500: loss 74.765429
iteration 900 / 1500: loss 73.964236
iteration 1000 / 1500: loss 73.413998
iteration 1100 / 1500: loss 72.534978
iteration 1200 / 1500: loss 71.716934
iteration 1300 / 1500: loss 70.941121
iteration 1400 / 1500: loss 69.865072
iteration 0 / 1500: loss 1557.415465
iteration 100 / 1500: loss 209.466521
iteration 200 / 1500: loss 29.864776
iteration 300 / 1500: loss 5.835391
iteration 400 / 1500: loss 2.652625
iteration 500 / 1500: loss 2.209007
iteration 600 / 1500: loss 2.196694
iteration 700 / 1500: loss 2.164345
iteration 800 / 1500: loss 2.149214
iteration 900 / 1500: loss 2.169095
iteration 1000 / 1500: loss 2.111574
iteration 1100 / 1500: loss 2.160658
iteration 1200 / 1500: loss 2.142117
iteration 1300 / 1500: loss 2.157616
iteration 1400 / 1500: loss 2.107820
iteration 0 / 1500: loss 776.889962
iteration 100 / 1500: loss 285.086347
iteration 200 / 1500: loss 105.480846
iteration 300 / 1500: loss 39.922879
iteration 400 / 1500: loss 15.891535
iteration 500 / 1500: loss 7.210845
iteration 600 / 1500: loss 3.908065
iteration 700 / 1500: loss 2.766412
iteration 800 / 1500: loss 2.289649
iteration 900 / 1500: loss 2.227650
iteration 1000 / 1500: loss 2.115463
iteration 1100 / 1500: loss 2.140612
iteration 1200 / 1500: loss 2.127761
iteration 1300 / 1500: loss 2.082787
iteration 1400 / 1500: loss 2.092909
```

```
iteration 0 / 1500: loss 314.541545
iteration 100 / 1500: loss 209.453621
iteration 200 / 1500: loss 140.641694
iteration 300 / 1500: loss 94.685744
iteration 400 / 1500: loss 63.853159
iteration 500 / 1500: loss 43.389303
iteration 600 / 1500: loss 29.782184
iteration 700 / 1500: loss 20.533121
iteration 800 / 1500: loss 14.429571
iteration 900 / 1500: loss 10.248729
iteration 1000 / 1500: loss 7.575195
iteration 1100 / 1500: loss 5.627953
iteration 1200 / 1500: loss 4.533996
iteration 1300 / 1500: loss 3.742923
iteration 1400 / 1500: loss 3.062863
iteration 0 / 1500: loss 159.569763
iteration 100 / 1500: loss 129.527120
iteration 200 / 1500: loss 105.915556
iteration 300 / 1500: loss 86.584633
iteration 400 / 1500: loss 71.319570
iteration 500 / 1500: loss 58.565172
iteration 600 / 1500: loss 48.112283
iteration 700 / 1500: loss 39.778463
iteration 800 / 1500: loss 32.847740
iteration 900 / 1500: loss 27.183992
iteration 1000 / 1500: loss 22.513701
iteration 1100 / 1500: loss 18.842620
iteration 1200 / 1500: loss 15.706334
iteration 1300 / 1500: loss 12.997266
iteration 1400 / 1500: loss 11.199775
iteration 0 / 1500: loss 82.479184
iteration 100 / 1500: loss 73.483255
iteration 200 / 1500: loss 66.246568
iteration 300 / 1500: loss 59.847073
iteration 400 / 1500: loss 54.120392
iteration 500 / 1500: loss 49.204425
iteration 600 / 1500: loss 44.565829
iteration 700 / 1500: loss 40.376900
iteration 800 / 1500: loss 36.549675
iteration 900 / 1500: loss 33.199731
iteration 1000 / 1500: loss 30.269645
iteration 1100 / 1500: loss 27.466278
iteration 1200 / 1500: loss 24.984394
iteration 1300 / 1500: loss 22.874199
iteration 1400 / 1500: loss 20.781651
iteration 0 / 1500: loss 1553.659980
iteration 100 / 1500: loss 2.131808
iteration 200 / 1500: loss 2.130302
```

```
iteration 300 / 1500: loss 2.164673
iteration 400 / 1500: loss 2.107973
iteration 500 / 1500: loss 2.123243
iteration 600 / 1500: loss 2.126238
iteration 700 / 1500: loss 2.153073
iteration 800 / 1500: loss 2.123970
iteration 900 / 1500: loss 2.135975
iteration 1000 / 1500: loss 2.182297
iteration 1100 / 1500: loss 2.178530
iteration 1200 / 1500: loss 2.151706
iteration 1300 / 1500: loss 2.218359
iteration 1400 / 1500: loss 2.132054
iteration 0 / 1500: loss 762.664333
iteration 100 / 1500: loss 2.092271
iteration 200 / 1500: loss 2.124408
iteration 300 / 1500: loss 2.112782
iteration 400 / 1500: loss 2.086238
iteration 500 / 1500: loss 2.091080
iteration 600 / 1500: loss 2.113308
iteration 700 / 1500: loss 2.081699
iteration 800 / 1500: loss 2.114679
iteration 900 / 1500: loss 2.027393
iteration 1000 / 1500: loss 2.144904
iteration 1100 / 1500: loss 2.080689
iteration 1200 / 1500: loss 2.154548
iteration 1300 / 1500: loss 2.109498
iteration 1400 / 1500: loss 2.060561
iteration 0 / 1500: loss 313.321300
iteration 100 / 1500: loss 7.368321
iteration 200 / 1500: loss 2.043654
iteration 300 / 1500: loss 2.063335
iteration 400 / 1500: loss 2.011212
iteration 500 / 1500: loss 2.017373
iteration 600 / 1500: loss 2.022572
iteration 700 / 1500: loss 2.056208
iteration 800 / 1500: loss 1.983082
iteration 900 / 1500: loss 2.002594
iteration 1000 / 1500: loss 2.065820
iteration 1100 / 1500: loss 2.143037
iteration 1200 / 1500: loss 1.982392
iteration 1300 / 1500: loss 2.015351
iteration 1400 / 1500: loss 2.003035
iteration 0 / 1500: loss 157.742978
iteration 100 / 1500: loss 22.180146
iteration 200 / 1500: loss 4.568277
iteration 300 / 1500: loss 2.242169
iteration 400 / 1500: loss 1.984128
iteration 500 / 1500: loss 1.936582
```

```
iteration 600 / 1500: loss 1.964431
iteration 700 / 1500: loss 1.946564
iteration 800 / 1500: loss 1.964017
iteration 900 / 1500: loss 2.053394
iteration 1000 / 1500: loss 1.920864
iteration 1100 / 1500: loss 1.967299
iteration 1200 / 1500: loss 2.088673
iteration 1300 / 1500: loss 1.978397
iteration 1400 / 1500: loss 2.014797
iteration 0 / 1500: loss 84.996554
iteration 100 / 1500: loss 30.587505
iteration 200 / 1500: loss 12.283805
iteration 300 / 1500: loss 5.715657
iteration 400 / 1500: loss 3.276137
iteration 500 / 1500: loss 2.366174
iteration 600 / 1500: loss 2.107187
iteration 700 / 1500: loss 2.014012
iteration 800 / 1500: loss 1.947572
iteration 900 / 1500: loss 1.908567
iteration 1000 / 1500: loss 1.842313
iteration 1100 / 1500: loss 2.003222
iteration 1200 / 1500: loss 1.858575
iteration 1300 / 1500: loss 1.858533
iteration 1400 / 1500: loss 1.811365
iteration 0 / 1500: loss 1549.950727
iteration 100 / 1500: loss 17.340026
iteration 200 / 1500: loss 16.167477
iteration 300 / 1500: loss 19.970590
iteration 400 / 1500: loss 16.350147
iteration 500 / 1500: loss 16.820714
iteration 600 / 1500: loss 17.907856
iteration 700 / 1500: loss 16.399238
iteration 800 / 1500: loss 15.202096
iteration 900 / 1500: loss 16.980061
iteration 1000 / 1500: loss 15.968075
iteration 1100 / 1500: loss 17.146955
iteration 1200 / 1500: loss 16.827579
iteration 1300 / 1500: loss 15.058586
iteration 1400 / 1500: loss 16.973222
iteration 0 / 1500: loss 765.798550
iteration 100 / 1500: loss 7.325224
iteration 200 / 1500: loss 8.598875
iteration 300 / 1500: loss 7.658884
iteration 400 / 1500: loss 7.565940
iteration 500 / 1500: loss 6.379633
iteration 600 / 1500: loss 6.982341
iteration 700 / 1500: loss 6.114895
iteration 800 / 1500: loss 6.241790
```

```
iteration 900 / 1500: loss 7.560435
iteration 1000 / 1500: loss 8.351801
iteration 1100 / 1500: loss 9.178528
iteration 1200 / 1500: loss 8.519503
iteration 1300 / 1500: loss 6.944554
iteration 1400 / 1500: loss 7.878553
iteration 0 / 1500: loss 315.853805
iteration 100 / 1500: loss 3.473608
iteration 200 / 1500: loss 3.680973
iteration 300 / 1500: loss 5.332848
iteration 400 / 1500: loss 3.978549
iteration 500 / 1500: loss 3.004498
iteration 600 / 1500: loss 4.439482
iteration 700 / 1500: loss 4.218646
iteration 800 / 1500: loss 3.933495
iteration 900 / 1500: loss 5.361097
iteration 1000 / 1500: loss 7.014121
iteration 1100 / 1500: loss 6.395994
iteration 1200 / 1500: loss 4.667159
iteration 1300 / 1500: loss 4.852414
iteration 1400 / 1500: loss 5.148905
iteration 0 / 1500: loss 159.345595
iteration 100 / 1500: loss 3.070365
iteration 200 / 1500: loss 4.056032
iteration 300 / 1500: loss 5.048136
iteration 400 / 1500: loss 4.256978
iteration 500 / 1500: loss 4.148021
iteration 600 / 1500: loss 3.845829
iteration 700 / 1500: loss 4.365163
iteration 800 / 1500: loss 4.653680
iteration 900 / 1500: loss 3.858860
iteration 1000 / 1500: loss 4.276107
iteration 1100 / 1500: loss 4.895707
iteration 1200 / 1500: loss 3.324174
iteration 1300 / 1500: loss 4.199189
iteration 1400 / 1500: loss 5.842314
iteration 0 / 1500: loss 83.762353
iteration 100 / 1500: loss 3.273397
iteration 200 / 1500: loss 3.447329
iteration 300 / 1500: loss 3.828118
iteration 400 / 1500: loss 5.016295
iteration 500 / 1500: loss 3.389708
iteration 600 / 1500: loss 4.751559
iteration 700 / 1500: loss 3.839793
iteration 800 / 1500: loss 3.892695
iteration 900 / 1500: loss 2.955925
iteration 1000 / 1500: loss 4.561239
iteration 1100 / 1500: loss 3.212930
```

```
iteration 1200 / 1500: loss 3.132344
iteration 1300 / 1500: loss 3.537956
iteration 1400 / 1500: loss 3.821968
lr 1.000000e-09 reg 2.500000e+03 train accuracy: 0.118980 val accuracy: 0.123000
lr 1.000000e-09 reg 5.000000e+03 train accuracy: 0.104449 val accuracy: 0.103000
lr 1.000000e-09 reg 1.000000e+04 train accuracy: 0.124286 val accuracy: 0.112000
lr 1.000000e-09 reg 2.500000e+04 train accuracy: 0.099245 val accuracy: 0.110000
lr 1.000000e-09 reg 5.000000e+04 train accuracy: 0.094388 val accuracy: 0.082000
lr 1.000000e-08 reg 2.500000e+03 train accuracy: 0.148041 val accuracy: 0.148000
lr 1.000000e-08 reg 5.000000e+03 train accuracy: 0.145449 val accuracy: 0.138000
lr 1.000000e-08 reg 1.000000e+04 train accuracy: 0.162265 val accuracy: 0.166000
lr 1.000000e-08 reg 2.500000e+04 train accuracy: 0.172796 val accuracy: 0.191000
lr 1.000000e-08 reg 5.000000e+04 train accuracy: 0.210224 val accuracy: 0.219000
lr 1.000000e-07 reg 2.500000e+03 train accuracy: 0.290857 val accuracy: 0.306000
lr 1.000000e-07 reg 5.000000e+03 train accuracy: 0.334061 val accuracy: 0.337000
lr 1.000000e-07 reg 1.000000e+04 train accuracy: 0.354408 val accuracy: 0.375000
lr 1.000000e-07 reg 2.500000e+04 train accuracy: 0.326653 val accuracy: 0.337000
lr 1.000000e-07 reg 5.000000e+04 train accuracy: 0.306878 val accuracy: 0.318000
lr 1.000000e-06 reg 2.500000e+03 train accuracy: 0.382898 val accuracy: 0.397000
lr 1.000000e-06 reg 5.000000e+03 train accuracy: 0.368286 val accuracy: 0.360000
lr 1.000000e-06 reg 1.000000e+04 train accuracy: 0.354531 val accuracy: 0.355000
lr 1.000000e-06 reg 2.500000e+04 train accuracy: 0.323388 val accuracy: 0.332000
lr 1.000000e-06 reg 5.000000e+04 train accuracy: 0.292020 val accuracy: 0.296000
lr 1.000000e-05 reg 2.500000e+03 train accuracy: 0.236408 val accuracy: 0.228000
lr 1.000000e-05 reg 5.000000e+03 train accuracy: 0.224469 val accuracy: 0.220000
lr 1.000000e-05 reg 1.000000e+04 train accuracy: 0.175122 val accuracy: 0.186000
lr 1.000000e-05 reg 2.500000e+04 train accuracy: 0.137388 val accuracy: 0.131000
lr 1.000000e-05 reg 5.000000e+04 train accuracy: 0.098673 val accuracy: 0.097000
best validation accuracy achieved during cross-validation: 0.397000
```

```python
[77]:  # Visualize the cross-validation results
       import math
       import pdb

       # pdb.set_trace()

       x_scatter = [math.log10(x[0]) for x in results]
       y_scatter = [math.log10(x[1]) for x in results]

       # plot training accuracy
       marker_size = 100
       colors = [results[x][0] for x in results]
       plt.subplot(2, 1, 1)
       plt.tight_layout(pad=3)
       plt.scatter(x_scatter, y_scatter, marker_size, c=colors, cmap=plt.cm.coolwarm)
       plt.colorbar()
       plt.xlabel('log learning rate')
```

```python
plt.ylabel('log regularization strength')
plt.title('CIFAR-10 training accuracy')

# plot validation accuracy
colors = [results[x][1] for x in results] # default size of markers is 20
plt.subplot(2, 1, 2)
plt.scatter(x_scatter, y_scatter, marker_size, c=colors, cmap=plt.cm.coolwarm)
plt.colorbar()
plt.xlabel('log learning rate')
plt.ylabel('log regularization strength')
plt.title('CIFAR-10 validation accuracy')
plt.show()
```
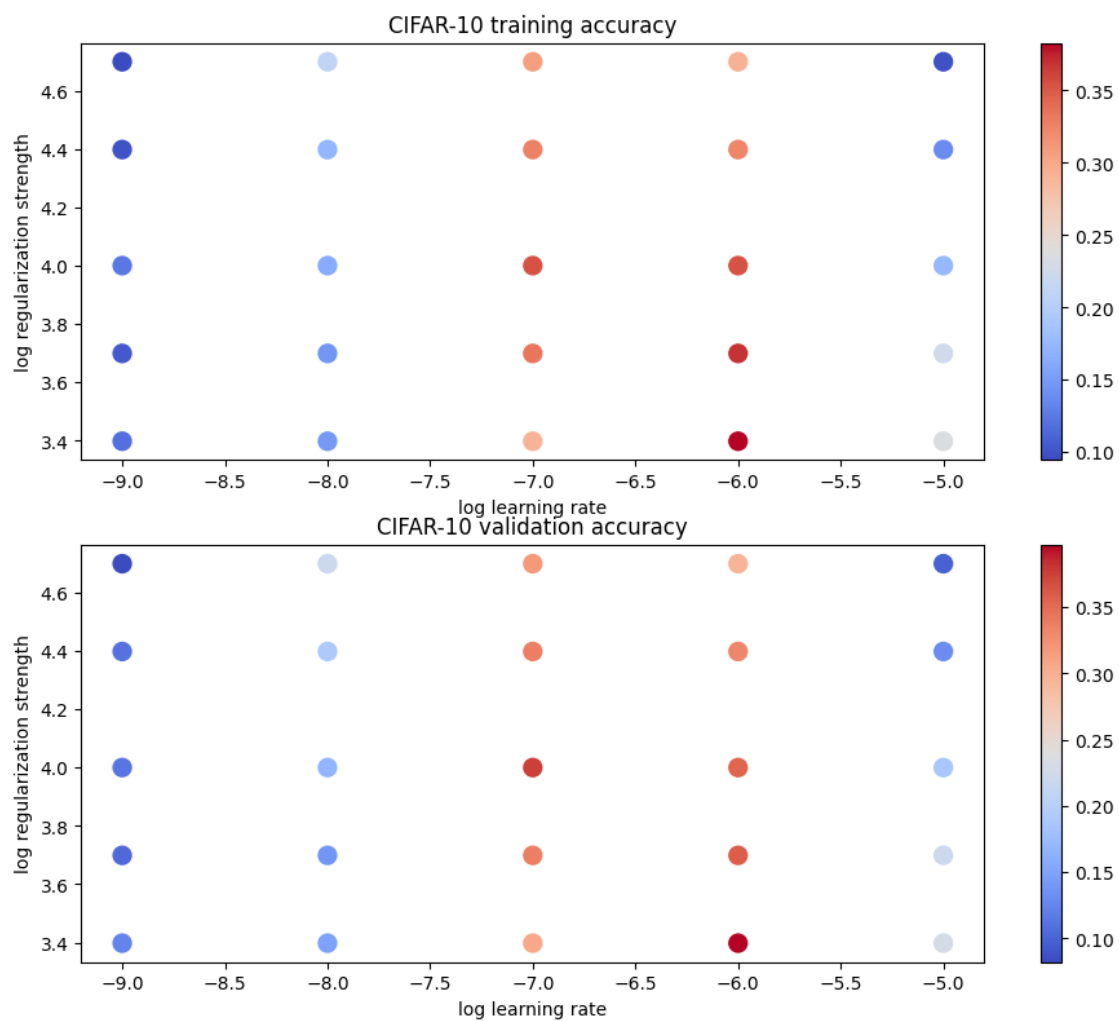


```python
[78]: # Evaluate the best softmax on test set
y_test_pred = best_softmax.predict(X_test)
```

```
test_accuracy = np.mean(y_test == y_test_pred)
print('Softmax classifier on raw pixels final test set accuracy: %f' %
  ↪test_accuracy)
```

Softmax classifier on raw pixels final test set accuracy: 0.379000

[79]:
```
# Save best softmax model
best_softmax.save("best_softmax.npy")
```

best_softmax.npy saved.

[80]:
```
# Visualize the learned weights for each class.
# Depending on your choice of learning rate and regularization strength, these
  ↪may
# or may not be nice to look at.
w = best_softmax.W[:-1,:] # strip out the bias
w = w.reshape(32, 32, 3, 10)
w_min, w_max = np.min(w), np.max(w)
classes = ['plane', 'car', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse',
  ↪'ship', 'truck']
for i in range(10):
    plt.subplot(2, 5, i + 1)

    # Rescale the weights to be between 0 and 255
    wimg = 255.0 * (w[:, :, :, i].squeeze() - w_min) / (w_max - w_min)
    plt.imshow(wimg.astype('uint8'))
    plt.axis('off')
    plt.title(classes[i])
```

plane	car	bird	cat	deer

dog	frog	horse	ship	truck

Inline question 3

Describe what your visualized Softmax classifier weights look like, and offer a brief explanation for why they look the way they do.

YourAnswer : *they look like a blurry color template of the class (for each class). the reason this happens is because the linear classifiers learns a single learned average for each class. to maximize the dot product for all images in a class we would want the weights to be as close as possible to the input data, therefore - it minimizes the error by effectively learning the pixel average (using SGD) of each pixel in the training examples (per class), creating a blurry color representation*

Inline Question 4 - *True or False*

Suppose the overall training loss is defined as the sum of the per-datapoint loss over all training examples. It is possible to add a new datapoint to a training set that would change the softmax loss, but leave the SVM loss unchanged.

YourAnswer : True

YourExplanation : the loss function of the SVM is: $L_i = \sum_{j \neq y_i} \max(0, s_j - s_{y_i} + \Delta)$ meaning if $s_{y_i} \geq s_j + \Delta$ we snap to zero. but with softmax, which has the loss: $L_i = -\log\left(\frac{e^{s_{y_i}}}{\sum_k e^{s_k}}\right)$, where we turn the predictions into probabilities, meaning for it to abselute zero we would need $s_{y_i} \to \infty$ and even then it would be a limit and not a absulote zero, and since we are in the "world" of computers, who handle finite numbers, this would never actually be zero (and always will be a bit lower then 1, even in the "limit")

[80]:

two_layer_net

December 18, 2025

```
[1]: # This mounts your Google Drive to the Colab VM.
     from google.colab import drive
     drive.mount('/content/drive')

     # TODO: Enter the foldername in your Drive where you have saved the unzipped
     # assignment folder, e.g. 'icv83551/assignments/assignment1/'
     FOLDERNAME = 'icv83551/assignments/assignment1/'
     assert FOLDERNAME is not None, "[!] Enter the foldername."

     # Now that we've mounted your Drive, this ensures that
     # the Python interpreter of the Colab VM can load
     # python files from within it.
     import sys
     sys.path.append('/content/drive/My Drive/{}'.format(FOLDERNAME))

     # This downloads the CIFAR-10 dataset to your Drive
     # if it doesn't already exist.
     %cd /content/drive/My\ Drive/$FOLDERNAME/icv83551/datasets/
     !bash get_datasets.sh
     %cd /content/drive/My\ Drive/$FOLDERNAME
```

```
Mounted at /content/drive
/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/datasets
/content/drive/My Drive/icv83551/assignments/assignment1
```

1 Fully-Connected Neural Nets

In this exercise we will implement fully-connected networks using a modular approach. For each layer we will implement a `forward` and a `backward` function. The `forward` function will receive inputs, weights, and other parameters and will return both an output and a `cache` object storing data needed for the backward pass, like this:

```
def layer_forward(x, w):
  """ Receive inputs x and weights w """
  # Do some computations ...
  z = # ... some intermediate value
  # Do some more computations ...
  out = # the output
```

1

```
    cache = (x, w, z, out) # Values we need to compute gradients

    return out, cache
```

The backward pass will receive upstream derivatives and the `cache` object, and will return gradients with respect to the inputs and weights, like this:

```
def layer_backward(dout, cache):
  """
  Receive dout (derivative of loss with respect to outputs) and cache,
  and compute derivative with respect to inputs.
  """
  # Unpack cache values
  x, w, z, out = cache

  # Use values in cache to compute derivatives
  dx = # Derivative of loss with respect to x
  dw = # Derivative of loss with respect to w

  return dx, dw
```

After implementing a bunch of layers this way, we will be able to easily combine them to build classifiers with different architectures.

```
[2]: # As usual, a bit of setup
from __future__ import print_function
import time
import numpy as np
import matplotlib.pyplot as plt
from icv83551.classifiers.fc_net import *
from icv83551.data_utils import get_CIFAR10_data
from icv83551.gradient_check import eval_numerical_gradient,␣
 ↪eval_numerical_gradient_array
from icv83551.solver import Solver

%matplotlib inline
plt.rcParams['figure.figsize'] = (10.0, 8.0) # set default size of plots
plt.rcParams['image.interpolation'] = 'nearest'
plt.rcParams['image.cmap'] = 'gray'

#%load_ext autoreload
#%autoreload 2

def rel_error(x, y):
  """ returns relative error """
  return np.max(np.abs(x - y) / (np.maximum(1e-8, np.abs(x) + np.abs(y))))
```

```
[3]: # Load the (preprocessed) CIFAR10 data.

     data = get_CIFAR10_data()
     for k, v in list(data.items()):
       print(('%s: ' % k, v.shape))
```

```
('X_train: ', (49000, 3, 32, 32))
('y_train: ', (49000,))
('X_val: ', (1000, 3, 32, 32))
('y_val: ', (1000,))
('X_test: ', (1000, 3, 32, 32))
('y_test: ', (1000,))
```

2 Affine layer: forward

Open the file `icv83551/layers.py` and implement the `affine_forward` function.

Once you are done you can test your implementaion by running the following:

```
[4]: # Test the affine_forward function

     num_inputs = 2
     input_shape = (4, 5, 6)
     output_dim = 3

     input_size = num_inputs * np.prod(input_shape)
     weight_size = output_dim * np.prod(input_shape)

     x = np.linspace(-0.1, 0.5, num=input_size).reshape(num_inputs, *input_shape)
     w = np.linspace(-0.2, 0.3, num=weight_size).reshape(np.prod(input_shape),
       ↪output_dim)
     b = np.linspace(-0.3, 0.1, num=output_dim)

     out, _ = affine_forward(x, w, b)
     correct_out = np.array([[ 1.49834967,  1.70660132,  1.91485297],
                             [ 3.25553199,  3.5141327,   3.77273342]])

     # Compare your output with ours. The error should be around e-9 or less.
     print('Testing affine_forward function:')
     print('difference: ', rel_error(out, correct_out))
```

```
Testing affine_forward function:
difference:  9.769849468192957e-10
```

3 Affine layer: backward

Now implement the `affine_backward` function and test your implementation using numeric gradient checking.

```
[5]: # Test the affine_backward function
     np.random.seed(231)
     x = np.random.randn(10, 2, 3)
     w = np.random.randn(6, 5)
     b = np.random.randn(5)
     dout = np.random.randn(10, 5)

     dx_num = eval_numerical_gradient_array(lambda x: affine_forward(x, w, b)[0], x,␣
       ↪dout)
     dw_num = eval_numerical_gradient_array(lambda w: affine_forward(x, w, b)[0], w,␣
       ↪dout)
     db_num = eval_numerical_gradient_array(lambda b: affine_forward(x, w, b)[0], b,␣
       ↪dout)

     _, cache = affine_forward(x, w, b)
     dx, dw, db = affine_backward(dout, cache)

     # The error should be around e-10 or less
     print('Testing affine_backward function:')
     print('dx error: ', rel_error(dx_num, dx))
     print('dw error: ', rel_error(dw_num, dw))
     print('db error: ', rel_error(db_num, db))
```

```
Testing affine_backward function:
dx error:  5.399100368651805e-11
dw error:  9.904211865398145e-11
db error:  2.4122867568119087e-11
```

4   ReLU activation: forward

Implement the forward pass for the ReLU activation function in the `relu_forward` function and test your implementation using the following:

```
[6]: # Test the relu_forward function

     x = np.linspace(-0.5, 0.5, num=12).reshape(3, 4)

     out, _ = relu_forward(x)
     correct_out = np.array([[ 0.,          0.,          0.,          0.,        ],
                             [ 0.,          0.,          0.04545455,  0.13636364,],
                             [ 0.22727273,  0.31818182,  0.40909091,  0.5,       ]])

     # Compare your output with ours. The error should be on the order of e-8
     print('Testing relu_forward function:')
     print('difference: ', rel_error(out, correct_out))
```

```
Testing relu_forward function:
difference:  4.999999798022158e-08
```

5   ReLU activation: backward

Now implement the backward pass for the ReLU activation function in the `relu_backward` function and test your implementation using numeric gradient checking:

```
[7]: np.random.seed(231)
     x = np.random.randn(10, 10)
     dout = np.random.randn(*x.shape)

     dx_num = eval_numerical_gradient_array(lambda x: relu_forward(x)[0], x, dout)

     _, cache = relu_forward(x)
     dx = relu_backward(dout, cache)

     # The error should be on the order of e-12
     print('Testing relu_backward function:')
     print('dx error: ', rel_error(dx_num, dx))
```

```
Testing relu_backward function:
dx error:  3.2756349136310288e-12
```

5.1   Inline Question 1:

We've only asked you to implement ReLU, but there are a number of different activation functions that one could use in neural networks, each with its pros and cons. In particular, an issue commonly seen with activation functions is getting zero (or close to zero) gradient flow during backpropagation. Which of the following activation functions have this problem? If you consider these functions in the one dimensional case, what types of input would lead to this behaviour? 1. Sigmoid 2. ReLU 3. Leaky ReLU

Your Answer : *1 & 2*

6   "Sandwich" layers

There are some common patterns of layers that are frequently used in neural nets. For example, affine layers are frequently followed by a ReLU nonlinearity. To make these common patterns easy, we define several convenience layers in the file `icv83551/layer_utils.py`.

For now take a look at the `affine_relu_forward` and `affine_relu_backward` functions, and run the following to numerically gradient check the backward pass:

```
[8]: from icv83551.layer_utils import affine_relu_forward, affine_relu_backward
     np.random.seed(231)
     x = np.random.randn(2, 3, 4)
     w = np.random.randn(12, 10)
     b = np.random.randn(10)
```

```
dout = np.random.randn(2, 10)


out, cache = affine_relu_forward(x, w, b)
dx, dw, db = affine_relu_backward(dout, cache)


dx_num = eval_numerical_gradient_array(lambda x: affine_relu_forward(x, w,␣
  ↪b)[0], x, dout)
dw_num = eval_numerical_gradient_array(lambda w: affine_relu_forward(x, w,␣
  ↪b)[0], w, dout)
db_num = eval_numerical_gradient_array(lambda b: affine_relu_forward(x, w,␣
  ↪b)[0], b, dout)


# Relative error should be around e-10 or less
print('Testing affine_relu_forward and affine_relu_backward:')
print('dx error: ', rel_error(dx_num, dx))
print('dw error: ', rel_error(dw_num, dw))
print('db error: ', rel_error(db_num, db))
```

```
Testing affine_relu_forward and affine_relu_backward:
dx error:  2.299579177309368e-11
dw error:  8.162011105764925e-11
db error:  7.826724021458994e-12
```

7 Loss layers: Softmax

Now implement the loss and gradient for softmax in the `softmax_loss` function in `icv83551/layers.py`. These should be similar to what you implemented in `icv83551/classifiers/softmax.py`. Other loss functions (e.g. `svm_loss`) can also be implemented in a modular way, however, it is not required for this assignment.

You can make sure that the implementations are correct by running the following:

```
[9]: np.random.seed(231)
num_classes, num_inputs = 10, 50
x = 0.001 * np.random.randn(num_inputs, num_classes)
y = np.random.randint(num_classes, size=num_inputs)


dx_num = eval_numerical_gradient(lambda x: softmax_loss(x, y)[0], x,␣
  ↪verbose=False)
loss, dx = softmax_loss(x, y)

# Test softmax_loss function. Loss should be close to 2.3 and dx error should␣
  ↪be around e-8
print('\nTesting softmax_loss:')
print('loss: ', loss)
print('dx error: ', rel_error(dx_num, dx))
```

```
Testing softmax_loss:
loss:   2.3025458445007376
dx error:   8.234144091578429e-09
```

8 Two-layer network

Open the file `icv83551/classifiers/fc_net.py` and complete the implementation of the `TwoLayerNet` class. Read through it to make sure you understand the API. You can run the cell below to test your implementation.

```python
[10]: np.random.seed(231)
      N, D, H, C = 3, 5, 50, 7
      X = np.random.randn(N, D)
      y = np.random.randint(C, size=N)

      std = 1e-3
      model = TwoLayerNet(input_dim=D, hidden_dim=H, num_classes=C, weight_scale=std)

      print('Testing initialization ... ')
      W1_std = abs(model.params['W1'].std() - std)
      b1 = model.params['b1']
      W2_std = abs(model.params['W2'].std() - std)
      b2 = model.params['b2']
      assert W1_std < std / 10, 'First layer weights do not seem right'
      assert np.all(b1 == 0), 'First layer biases do not seem right'
      assert W2_std < std / 10, 'Second layer weights do not seem right'
      assert np.all(b2 == 0), 'Second layer biases do not seem right'

      print('Testing test-time forward pass ... ')
      model.params['W1'] = np.linspace(-0.7, 0.3, num=D*H).reshape(D, H)
      model.params['b1'] = np.linspace(-0.1, 0.9, num=H)
      model.params['W2'] = np.linspace(-0.3, 0.4, num=H*C).reshape(H, C)
      model.params['b2'] = np.linspace(-0.9, 0.1, num=C)
      X = np.linspace(-5.5, 4.5, num=N*D).reshape(D, N).T
      scores = model.loss(X)
      correct_scores = np.asarray(
        [[11.53165108,  12.2917344,    13.05181771,  13.81190102,  14.57198434, 15.
      ↪33206765,  16.09215096],
         [12.05769098,  12.74614105,  13.43459113,  14.1230412,   14.81149128, 15.
      ↪49994135,  16.18839143],
         [12.58373087,  13.20054771,  13.81736455,  14.43418138,  15.05099822, 15.
      ↪66781506,  16.2846319 ]])
      scores_diff = np.abs(scores - correct_scores).sum()
      assert scores_diff < 1e-6, 'Problem with test-time forward pass'

      print('Testing training loss (no regularization)')
```

```
y = np.asarray([0, 5, 1])
loss, grads = model.loss(X, y)
correct_loss = 3.4702243556
assert abs(loss - correct_loss) < 1e-10, 'Problem with training-time loss'

model.reg = 1.0
loss, grads = model.loss(X, y)
correct_loss = 26.5948426952
assert abs(loss - correct_loss) < 1e-10, 'Problem with regularization loss'

# Errors should be around e-7 or less
for reg in [0.0, 0.7]:
  print('Running numeric gradient check with reg = ', reg)
  model.reg = reg
  loss, grads = model.loss(X, y)

  for name in sorted(grads):
    f = lambda _: model.loss(X, y)[0]
    grad_num = eval_numerical_gradient(f, model.params[name], verbose=False)
    print('%s relative error: %.2e' % (name, rel_error(grad_num, grads[name])))
```

```
Testing initialization …
Testing test-time forward pass …
Testing training loss (no regularization)
Running numeric gradient check with reg =  0.0
W1 relative error: 1.83e-08
W2 relative error: 3.20e-10
b1 relative error: 9.83e-09
b2 relative error: 4.33e-10
Running numeric gradient check with reg =  0.7
W1 relative error: 2.53e-07
W2 relative error: 2.85e-08
b1 relative error: 1.56e-08
b2 relative error: 9.09e-10
```

9 Solver

Open the file `icv83551/solver.py` and read through it to familiarize yourself with the API. After doing so, use a `Solver` instance to train a `TwoLayerNet` that achieves about **36%** accuracy on the validation set.

```
[11]: input_size = 32 * 32 * 3
      hidden_size = 50
      num_classes = 10
      model = TwoLayerNet(input_size, hidden_size, num_classes)
      solver = Solver(model,
                      data,
```

```
                update_rule='sgd',
                optim_config={'learning_rate': 1e-4},
                lr_decay=0.95,
                num_epochs=5,
                batch_size=200,
                print_every=100)
solver.train()
solver.check_accuracy(data['X_val'], data['y_val'])


################################################################################
# TODO: Use a Solver instance to train a TwoLayerNet that achieves about 36% #
# accuracy on the validation set.                                            #
################################################################################


################################################################################
#                            END OF YOUR CODE                                #
################################################################################
```

```
(Iteration 1 / 1225) loss: 2.301725
(Epoch 0 / 5) train acc: 0.145000; val_acc: 0.140000
(Iteration 101 / 1225) loss: 2.241923
(Iteration 201 / 1225) loss: 2.187425
(Epoch 1 / 5) train acc: 0.267000; val_acc: 0.243000
(Iteration 301 / 1225) loss: 2.056790
(Iteration 401 / 1225) loss: 1.937978
(Epoch 2 / 5) train acc: 0.294000; val_acc: 0.303000
(Iteration 501 / 1225) loss: 1.924555
(Iteration 601 / 1225) loss: 1.933743
(Iteration 701 / 1225) loss: 1.832777
(Epoch 3 / 5) train acc: 0.336000; val_acc: 0.315000
(Iteration 801 / 1225) loss: 1.960827
(Iteration 901 / 1225) loss: 1.832752
(Epoch 4 / 5) train acc: 0.340000; val_acc: 0.350000
(Iteration 1001 / 1225) loss: 1.739182
(Iteration 1101 / 1225) loss: 1.940517
(Iteration 1201 / 1225) loss: 1.848443
(Epoch 5 / 5) train acc: 0.355000; val_acc: 0.373000
```

[11]: np.float64(0.373)

10   Debug the training

With the default parameters we provided above, you should get a validation accuracy of about 0.36 on the validation set. This isn't very good.

One strategy for getting insight into what's wrong is to plot the loss function and the accuracies on the training and validation sets during optimization.

Another strategy is to visualize the weights that were learned in the first layer of the network. In most neural networks trained on visual data, the first layer weights typically show some visible structure when visualized.

```
[12]:  # Run this cell to visualize training loss and train / val accuracy

       plt.subplot(2, 1, 1)
       plt.title('Training loss')
       plt.plot(solver.loss_history, 'o')
       plt.xlabel('Iteration')

       plt.subplot(2, 1, 2)
       plt.title('Accuracy')
       plt.plot(solver.train_acc_history, '-o', label='train')
       plt.plot(solver.val_acc_history, '-o', label='val')
       plt.plot([0.5] * len(solver.val_acc_history), 'k--')
       plt.xlabel('Epoch')
       plt.legend(loc='lower right')
       plt.gcf().set_size_inches(15, 12)
       plt.show()
```
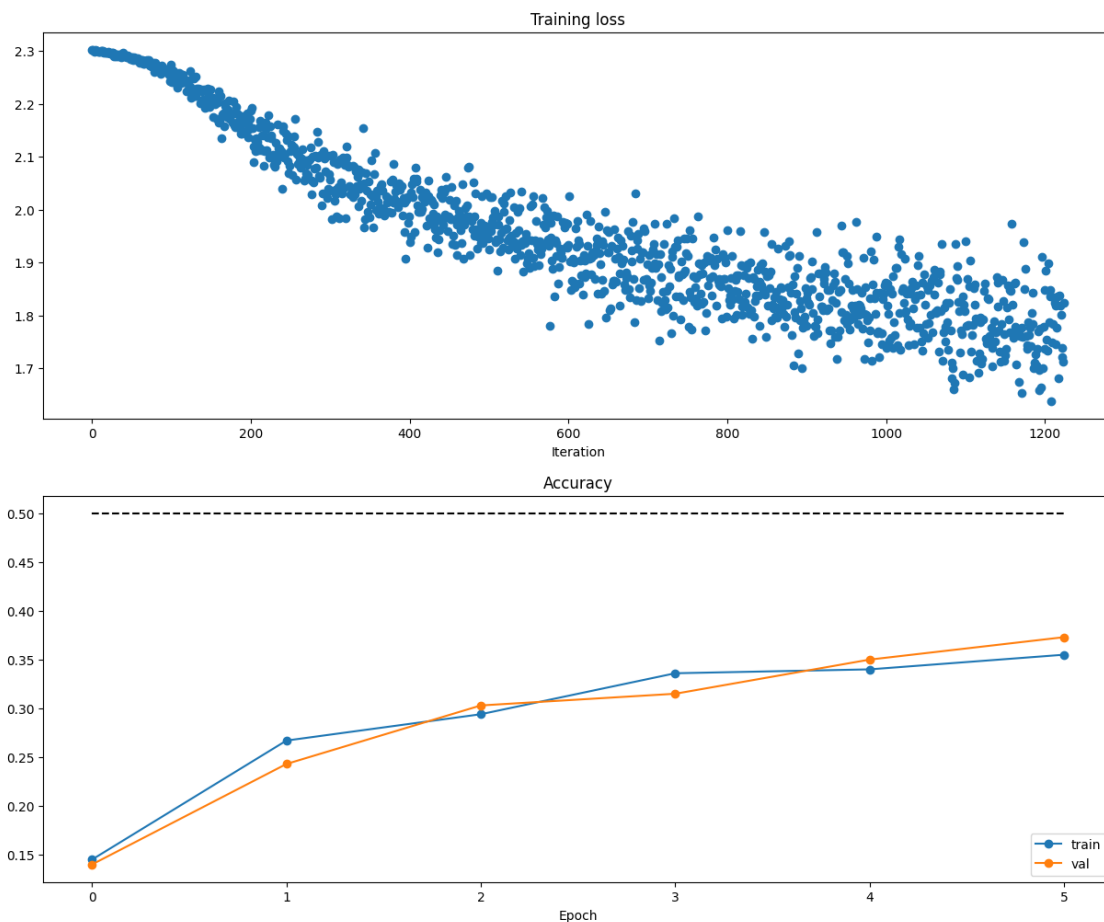
```
[13]:  from icv83551.vis_utils import visualize_grid

       # Visualize the weights of the network

       def show_net_weights(net):
           W1 = net.params['W1']
           W1 = W1.reshape(3, 32, 32, -1).transpose(3, 1, 2, 0)
           plt.imshow(visualize_grid(W1, padding=3).astype('uint8'))
           plt.gca().axis('off')
           plt.show()

       show_net_weights(model)
```
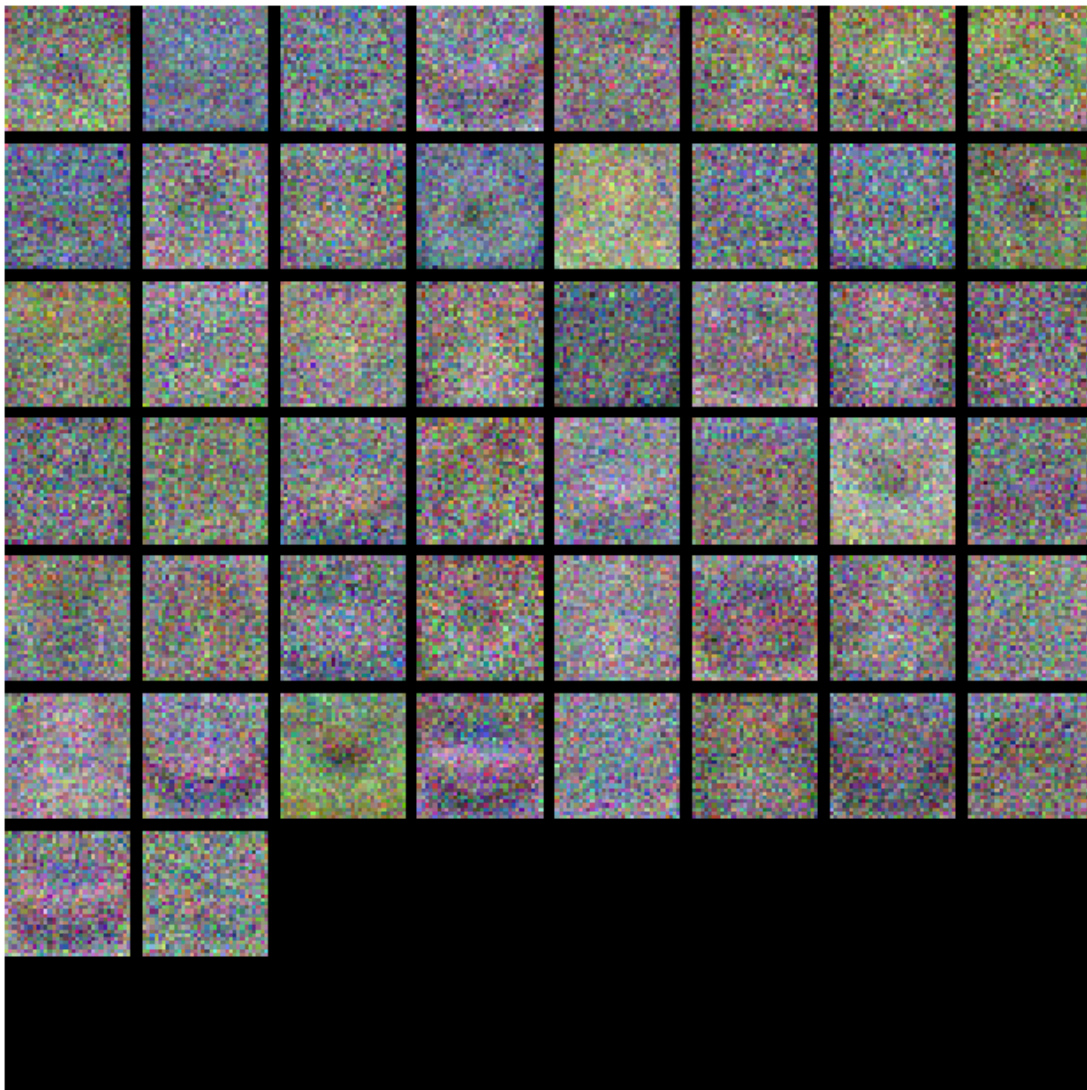
11 Tune your hyperparameters

What's wrong?. Looking at the visualizations above, we see that the loss is decreasing more or less linearly, which seems to suggest that the learning rate may be too low. Moreover, there is no gap between the training and validation accuracy, suggesting that the model we used has low capacity, and that we should increase its size. On the other hand, with a very large model we would expect to see more overfitting, which would manifest itself as a very large gap between the training and validation accuracy.

Tuning. Tuning the hyperparameters and developing intuition for how they affect the final performance is a large part of using Neural Networks, so we want you to get a lot of practice. Below, you should experiment with different values of the various hyperparameters, including hidden layer size, learning rate, numer of training epochs, and regularization strength. You might also consider tuning the learning rate decay, but you should be able to get good performance using the default value.

Approximate results. You should be aim to achieve a classification accuracy of greater than 48% on the validation set. Our best network gets over 52% on the validation set.

Experiment: You goal in this exercise is to get as good of a result on CIFAR-10 as you can (52% could serve as a reference), with a fully-connected Neural Network. Feel free implement your own techniques (e.g. PCA to reduce dimensionality, or adding dropout, or adding features to the solver, etc.).

```python
[14]: import itertools
best_model = None
highest_acc = 0
learningRate = [5*1e-4, 5*1e-3, 1e-4]
hiddenSize = [100, 200, 300]
eppochs = [4,5,6]
reg_st =[0.3,0.4,0.5,0.6]
batch_size = [100,200,300]


##############################################################################
# TODO: Tune hyperparameters using the validation set. Store your best trained ⎵
 ↪#
# model in best_model.                                                          ⎵
 ↪#
#                                                                               ⎵
 ↪#
# To help debug your network, it may help to use visualizations similar to the ⎵
 ↪#
# ones we used above; these visualizations will have significant qualitative    ⎵
 ↪#
```

```
# differences from the ones we saw above for the poorly tuned network.       ␣
  ↪#
#                                                                             ␣
  ↪#
# Tweaking hyperparameters by hand can be fun, but you might find it useful to ␣
  ↪#
# write code to sweep through possible combinations of hyperparameters        ␣
  ↪#
# automatically like we did on thexs previous exercises.                      ␣
  ↪#
################################################################################

for lr, hs, ep, reg_s, bs in itertools.product(
    learningRate, hiddenSize, eppochs, reg_st, batch_size
):
  model = TwoLayerNet(input_size, hs, num_classes, reg=reg_s)
  solver = Solver(model,
                  data,
                  update_rule='sgd',
                  optim_config={'learning_rate': lr},
                  lr_decay=0.95,
                  num_epochs=ep,
                  batch_size=bs,
                  print_every=100)
  solver.train()
  acc = solver.check_accuracy(data['X_val'], data['y_val'])
  if acc > highest_acc:
    highest_acc = acc
    best_model = model
################################################################################
#                              END OF YOUR CODE                               #
################################################################################
```

```
(Iteration 1 / 1960) loss: 2.349709
(Epoch 0 / 4) train acc: 0.114000; val_acc: 0.127000
(Iteration 101 / 1960) loss: 2.027995
(Iteration 201 / 1960) loss: 1.810853
(Iteration 301 / 1960) loss: 1.858742
(Iteration 401 / 1960) loss: 1.802049
(Epoch 1 / 4) train acc: 0.418000; val_acc: 0.442000
(Iteration 501 / 1960) loss: 1.749632
(Iteration 601 / 1960) loss: 1.657398
(Iteration 701 / 1960) loss: 1.627252
(Iteration 801 / 1960) loss: 1.656986
(Iteration 901 / 1960) loss: 1.732236
(Epoch 2 / 4) train acc: 0.482000; val_acc: 0.457000
(Iteration 1001 / 1960) loss: 1.726849
```

```
(Iteration 1101 / 1960) loss: 1.439827
(Iteration 1201 / 1960) loss: 1.501933
(Iteration 1301 / 1960) loss: 1.387258
(Iteration 1401 / 1960) loss: 1.671270
(Epoch 3 / 4) train acc: 0.488000; val_acc: 0.489000
(Iteration 1501 / 1960) loss: 1.495877
(Iteration 1601 / 1960) loss: 1.456625
(Iteration 1701 / 1960) loss: 1.446837
(Iteration 1801 / 1960) loss: 1.368949
(Iteration 1901 / 1960) loss: 1.354377
(Epoch 4 / 4) train acc: 0.501000; val_acc: 0.489000
(Iteration 1 / 980) loss: 2.349545
(Epoch 0 / 4) train acc: 0.123000; val_acc: 0.137000
(Iteration 101 / 980) loss: 1.953257
(Iteration 201 / 980) loss: 1.846551
(Epoch 1 / 4) train acc: 0.381000; val_acc: 0.395000
(Iteration 301 / 980) loss: 1.833981
(Iteration 401 / 980) loss: 1.714438
(Epoch 2 / 4) train acc: 0.415000; val_acc: 0.429000
(Iteration 501 / 980) loss: 1.711217
(Iteration 601 / 980) loss: 1.599932
(Iteration 701 / 980) loss: 1.624269
(Epoch 3 / 4) train acc: 0.459000; val_acc: 0.466000
(Iteration 801 / 980) loss: 1.556703
(Iteration 901 / 980) loss: 1.625575
(Epoch 4 / 4) train acc: 0.447000; val_acc: 0.476000
(Iteration 1 / 652) loss: 2.349426
(Epoch 0 / 4) train acc: 0.127000; val_acc: 0.127000
(Iteration 101 / 652) loss: 1.961926
(Epoch 1 / 4) train acc: 0.356000; val_acc: 0.348000
(Iteration 201 / 652) loss: 1.854768
(Iteration 301 / 652) loss: 1.701834
(Epoch 2 / 4) train acc: 0.405000; val_acc: 0.424000
(Iteration 401 / 652) loss: 1.662044
(Epoch 3 / 4) train acc: 0.441000; val_acc: 0.453000
(Iteration 501 / 652) loss: 1.610605
(Iteration 601 / 652) loss: 1.682130
(Epoch 4 / 4) train acc: 0.445000; val_acc: 0.462000
(Iteration 1 / 1960) loss: 2.364197
(Epoch 0 / 4) train acc: 0.122000; val_acc: 0.104000
(Iteration 101 / 1960) loss: 1.821332
(Iteration 201 / 1960) loss: 1.736783
(Iteration 301 / 1960) loss: 1.622907
(Iteration 401 / 1960) loss: 1.701360
(Epoch 1 / 4) train acc: 0.438000; val_acc: 0.431000
(Iteration 501 / 1960) loss: 1.669014
(Iteration 601 / 1960) loss: 1.528834
(Iteration 701 / 1960) loss: 1.598258
```

```
(Iteration 801 / 1960) loss: 1.608882
(Iteration 901 / 1960) loss: 1.527980
(Epoch 2 / 4) train acc: 0.482000; val_acc: 0.465000
(Iteration 1001 / 1960) loss: 1.661625
(Iteration 1101 / 1960) loss: 1.534955
(Iteration 1201 / 1960) loss: 1.565296
(Iteration 1301 / 1960) loss: 1.571512
(Iteration 1401 / 1960) loss: 1.623838
(Epoch 3 / 4) train acc: 0.489000; val_acc: 0.469000
(Iteration 1501 / 1960) loss: 1.380226
(Iteration 1601 / 1960) loss: 1.390883
(Iteration 1701 / 1960) loss: 1.598451
(Iteration 1801 / 1960) loss: 1.490944
(Iteration 1901 / 1960) loss: 1.585898
(Epoch 4 / 4) train acc: 0.523000; val_acc: 0.496000
(Iteration 1 / 980) loss: 2.366260
(Epoch 0 / 4) train acc: 0.093000; val_acc: 0.084000
(Iteration 101 / 980) loss: 2.011250
(Iteration 201 / 980) loss: 1.856663
(Epoch 1 / 4) train acc: 0.372000; val_acc: 0.390000
(Iteration 301 / 980) loss: 1.759662
(Iteration 401 / 980) loss: 1.747905
(Epoch 2 / 4) train acc: 0.451000; val_acc: 0.450000
(Iteration 501 / 980) loss: 1.799139
(Iteration 601 / 980) loss: 1.685797
(Iteration 701 / 980) loss: 1.539203
(Epoch 3 / 4) train acc: 0.467000; val_acc: 0.452000
(Iteration 801 / 980) loss: 1.600007
(Iteration 901 / 980) loss: 1.615151
(Epoch 4 / 4) train acc: 0.509000; val_acc: 0.454000
(Iteration 1 / 652) loss: 2.364587
(Epoch 0 / 4) train acc: 0.122000; val_acc: 0.103000
(Iteration 101 / 652) loss: 2.012285
(Epoch 1 / 4) train acc: 0.366000; val_acc: 0.358000
(Iteration 201 / 652) loss: 1.762787
(Iteration 301 / 652) loss: 1.778401
(Epoch 2 / 4) train acc: 0.388000; val_acc: 0.428000
(Iteration 401 / 652) loss: 1.704830
(Epoch 3 / 4) train acc: 0.414000; val_acc: 0.436000
(Iteration 501 / 652) loss: 1.731918
(Iteration 601 / 652) loss: 1.678911
(Epoch 4 / 4) train acc: 0.472000; val_acc: 0.462000
(Iteration 1 / 1960) loss: 2.382829
(Epoch 0 / 4) train acc: 0.124000; val_acc: 0.116000
(Iteration 101 / 1960) loss: 2.045953
(Iteration 201 / 1960) loss: 1.803125
(Iteration 301 / 1960) loss: 1.763269
(Iteration 401 / 1960) loss: 1.901430
```

```
(Epoch 1 / 4) train acc: 0.468000; val_acc: 0.436000
(Iteration 501 / 1960) loss: 1.601584
(Iteration 601 / 1960) loss: 1.705686
(Iteration 701 / 1960) loss: 1.573874
(Iteration 801 / 1960) loss: 1.561818
(Iteration 901 / 1960) loss: 1.571657
(Epoch 2 / 4) train acc: 0.471000; val_acc: 0.460000
(Iteration 1001 / 1960) loss: 1.617987
(Iteration 1101 / 1960) loss: 1.608900
(Iteration 1201 / 1960) loss: 1.597393
(Iteration 1301 / 1960) loss: 1.377159
(Iteration 1401 / 1960) loss: 1.629124
(Epoch 3 / 4) train acc: 0.501000; val_acc: 0.480000
(Iteration 1501 / 1960) loss: 1.388703
(Iteration 1601 / 1960) loss: 1.428193
(Iteration 1701 / 1960) loss: 1.484397
(Iteration 1801 / 1960) loss: 1.467317
(Iteration 1901 / 1960) loss: 1.539234
(Epoch 4 / 4) train acc: 0.546000; val_acc: 0.494000
(Iteration 1 / 980) loss: 2.378904
(Epoch 0 / 4) train acc: 0.107000; val_acc: 0.109000
(Iteration 101 / 980) loss: 2.047620
(Iteration 201 / 980) loss: 1.844433
(Epoch 1 / 4) train acc: 0.375000; val_acc: 0.384000
(Iteration 301 / 980) loss: 1.791438
(Iteration 401 / 980) loss: 1.613798
(Epoch 2 / 4) train acc: 0.451000; val_acc: 0.442000
(Iteration 501 / 980) loss: 1.759208
(Iteration 601 / 980) loss: 1.792614
(Iteration 701 / 980) loss: 1.687500
(Epoch 3 / 4) train acc: 0.465000; val_acc: 0.458000
(Iteration 801 / 980) loss: 1.659358
(Iteration 901 / 980) loss: 1.590272
(Epoch 4 / 4) train acc: 0.460000; val_acc: 0.459000
(Iteration 1 / 652) loss: 2.380135
(Epoch 0 / 4) train acc: 0.083000; val_acc: 0.089000
(Iteration 101 / 652) loss: 1.968799
(Epoch 1 / 4) train acc: 0.356000; val_acc: 0.371000
(Iteration 201 / 652) loss: 1.794823
(Iteration 301 / 652) loss: 1.746590
(Epoch 2 / 4) train acc: 0.420000; val_acc: 0.402000
(Iteration 401 / 652) loss: 1.722980
(Epoch 3 / 4) train acc: 0.447000; val_acc: 0.438000
(Iteration 501 / 652) loss: 1.680793
(Iteration 601 / 652) loss: 1.614198
(Epoch 4 / 4) train acc: 0.440000; val_acc: 0.463000
(Iteration 1 / 1960) loss: 2.397875
(Epoch 0 / 4) train acc: 0.129000; val_acc: 0.106000
```

```
(Iteration 101 / 1960) loss: 2.075188
(Iteration 201 / 1960) loss: 1.815514
(Iteration 301 / 1960) loss: 1.834976
(Iteration 401 / 1960) loss: 2.020702
(Epoch 1 / 4) train acc: 0.416000; val_acc: 0.444000
(Iteration 501 / 1960) loss: 1.763636
(Iteration 601 / 1960) loss: 1.565058
(Iteration 701 / 1960) loss: 1.552992
(Iteration 801 / 1960) loss: 1.735781
(Iteration 901 / 1960) loss: 1.586974
(Epoch 2 / 4) train acc: 0.476000; val_acc: 0.455000
(Iteration 1001 / 1960) loss: 1.589539
(Iteration 1101 / 1960) loss: 1.607988
(Iteration 1201 / 1960) loss: 1.728934
(Iteration 1301 / 1960) loss: 1.416312
(Iteration 1401 / 1960) loss: 1.653173
(Epoch 3 / 4) train acc: 0.502000; val_acc: 0.469000
(Iteration 1501 / 1960) loss: 1.364342
(Iteration 1601 / 1960) loss: 1.444178
(Iteration 1701 / 1960) loss: 1.481166
(Iteration 1801 / 1960) loss: 1.473068
(Iteration 1901 / 1960) loss: 1.561777
(Epoch 4 / 4) train acc: 0.504000; val_acc: 0.490000
(Iteration 1 / 980) loss: 2.393565
(Epoch 0 / 4) train acc: 0.149000; val_acc: 0.137000
(Iteration 101 / 980) loss: 2.035918
(Iteration 201 / 980) loss: 1.799849
(Epoch 1 / 4) train acc: 0.385000; val_acc: 0.387000
(Iteration 301 / 980) loss: 1.808421
(Iteration 401 / 980) loss: 1.800684
(Epoch 2 / 4) train acc: 0.416000; val_acc: 0.447000
(Iteration 501 / 980) loss: 1.718641
(Iteration 601 / 980) loss: 1.741613
(Iteration 701 / 980) loss: 1.683211
(Epoch 3 / 4) train acc: 0.478000; val_acc: 0.460000
(Iteration 801 / 980) loss: 1.528681
(Iteration 901 / 980) loss: 1.523596
(Epoch 4 / 4) train acc: 0.506000; val_acc: 0.462000
(Iteration 1 / 652) loss: 2.393930
(Epoch 0 / 4) train acc: 0.127000; val_acc: 0.136000
(Iteration 101 / 652) loss: 2.015130
(Epoch 1 / 4) train acc: 0.340000; val_acc: 0.356000
(Iteration 201 / 652) loss: 1.908222
(Iteration 301 / 652) loss: 1.735777
(Epoch 2 / 4) train acc: 0.426000; val_acc: 0.416000
(Iteration 401 / 652) loss: 1.687841
(Epoch 3 / 4) train acc: 0.440000; val_acc: 0.456000
(Iteration 501 / 652) loss: 1.769423
```

```
(Iteration 601 / 652) loss: 1.724162
(Epoch 4 / 4) train acc: 0.474000; val_acc: 0.465000
(Iteration 1 / 2450) loss: 2.351504
(Epoch 0 / 5) train acc: 0.090000; val_acc: 0.101000
(Iteration 101 / 2450) loss: 2.064953
(Iteration 201 / 2450) loss: 1.944838
(Iteration 301 / 2450) loss: 1.709412
(Iteration 401 / 2450) loss: 1.784895
(Epoch 1 / 5) train acc: 0.439000; val_acc: 0.415000
(Iteration 501 / 2450) loss: 1.667575
(Iteration 601 / 2450) loss: 1.582784
(Iteration 701 / 2450) loss: 1.598040
(Iteration 801 / 2450) loss: 1.640756
(Iteration 901 / 2450) loss: 1.467290
(Epoch 2 / 5) train acc: 0.505000; val_acc: 0.476000
(Iteration 1001 / 2450) loss: 1.578755
(Iteration 1101 / 2450) loss: 1.435212
(Iteration 1201 / 2450) loss: 1.440494
(Iteration 1301 / 2450) loss: 1.481910
(Iteration 1401 / 2450) loss: 1.640341
(Epoch 3 / 5) train acc: 0.507000; val_acc: 0.477000
(Iteration 1501 / 2450) loss: 1.383896
(Iteration 1601 / 2450) loss: 1.502480
(Iteration 1701 / 2450) loss: 1.558359
(Iteration 1801 / 2450) loss: 1.749256
(Iteration 1901 / 2450) loss: 1.366321
(Epoch 4 / 5) train acc: 0.521000; val_acc: 0.489000
(Iteration 2001 / 2450) loss: 1.467041
(Iteration 2101 / 2450) loss: 1.584367
(Iteration 2201 / 2450) loss: 1.657090
(Iteration 2301 / 2450) loss: 1.401217
(Iteration 2401 / 2450) loss: 1.259808
(Epoch 5 / 5) train acc: 0.533000; val_acc: 0.476000
(Iteration 1 / 1225) loss: 2.350229
(Epoch 0 / 5) train acc: 0.128000; val_acc: 0.159000
(Iteration 101 / 1225) loss: 2.011055
(Iteration 201 / 1225) loss: 1.850728
(Epoch 1 / 5) train acc: 0.364000; val_acc: 0.384000
(Iteration 301 / 1225) loss: 1.742569
(Iteration 401 / 1225) loss: 1.703972
(Epoch 2 / 5) train acc: 0.416000; val_acc: 0.427000
(Iteration 501 / 1225) loss: 1.648992
(Iteration 601 / 1225) loss: 1.627263
(Iteration 701 / 1225) loss: 1.694547
(Epoch 3 / 5) train acc: 0.460000; val_acc: 0.468000
(Iteration 801 / 1225) loss: 1.701229
(Iteration 901 / 1225) loss: 1.550885
(Epoch 4 / 5) train acc: 0.470000; val_acc: 0.459000
```

```
(Iteration 1001 / 1225) loss: 1.615063
(Iteration 1101 / 1225) loss: 1.612167
(Iteration 1201 / 1225) loss: 1.447337
(Epoch 5 / 5) train acc: 0.508000; val_acc: 0.488000
(Iteration 1 / 815) loss: 2.350885
(Epoch 0 / 5) train acc: 0.116000; val_acc: 0.100000
(Iteration 101 / 815) loss: 1.963046
(Epoch 1 / 5) train acc: 0.339000; val_acc: 0.358000
(Iteration 201 / 815) loss: 1.752721
(Iteration 301 / 815) loss: 1.878924
(Epoch 2 / 5) train acc: 0.407000; val_acc: 0.413000
(Iteration 401 / 815) loss: 1.738420
(Epoch 3 / 5) train acc: 0.419000; val_acc: 0.438000
(Iteration 501 / 815) loss: 1.683269
(Iteration 601 / 815) loss: 1.577500
(Epoch 4 / 5) train acc: 0.453000; val_acc: 0.448000
(Iteration 701 / 815) loss: 1.619758
(Iteration 801 / 815) loss: 1.610668
(Epoch 5 / 5) train acc: 0.461000; val_acc: 0.471000
(Iteration 1 / 2450) loss: 2.361803
(Epoch 0 / 5) train acc: 0.134000; val_acc: 0.147000
(Iteration 101 / 2450) loss: 1.941790
(Iteration 201 / 2450) loss: 1.820489
(Iteration 301 / 2450) loss: 1.772256
(Iteration 401 / 2450) loss: 1.720748
(Epoch 1 / 5) train acc: 0.428000; val_acc: 0.432000
(Iteration 501 / 2450) loss: 1.842984
(Iteration 601 / 2450) loss: 1.815234
(Iteration 701 / 2450) loss: 1.930787
(Iteration 801 / 2450) loss: 1.699931
(Iteration 901 / 2450) loss: 1.482557
(Epoch 2 / 5) train acc: 0.462000; val_acc: 0.458000
(Iteration 1001 / 2450) loss: 1.646994
(Iteration 1101 / 2450) loss: 1.622651
(Iteration 1201 / 2450) loss: 1.638794
(Iteration 1301 / 2450) loss: 1.585929
(Iteration 1401 / 2450) loss: 1.646450
(Epoch 3 / 5) train acc: 0.506000; val_acc: 0.474000
(Iteration 1501 / 2450) loss: 1.406922
(Iteration 1601 / 2450) loss: 1.613306
(Iteration 1701 / 2450) loss: 1.340028
(Iteration 1801 / 2450) loss: 1.602495
(Iteration 1901 / 2450) loss: 1.390807
(Epoch 4 / 5) train acc: 0.497000; val_acc: 0.499000
(Iteration 2001 / 2450) loss: 1.471477
(Iteration 2101 / 2450) loss: 1.343036
(Iteration 2201 / 2450) loss: 1.370085
(Iteration 2301 / 2450) loss: 1.349371
```

```
(Iteration 2401 / 2450) loss: 1.411307
(Epoch 5 / 5) train acc: 0.534000; val_acc: 0.496000
(Iteration 1 / 1225) loss: 2.365274
(Epoch 0 / 5) train acc: 0.078000; val_acc: 0.083000
(Iteration 101 / 1225) loss: 1.976609
(Iteration 201 / 1225) loss: 1.869426
(Epoch 1 / 5) train acc: 0.351000; val_acc: 0.381000
(Iteration 301 / 1225) loss: 1.775397
(Iteration 401 / 1225) loss: 1.722058
(Epoch 2 / 5) train acc: 0.439000; val_acc: 0.446000
(Iteration 501 / 1225) loss: 1.731688
(Iteration 601 / 1225) loss: 1.695779
(Iteration 701 / 1225) loss: 1.739830
(Epoch 3 / 5) train acc: 0.473000; val_acc: 0.453000
(Iteration 801 / 1225) loss: 1.615186
(Iteration 901 / 1225) loss: 1.536209
(Epoch 4 / 5) train acc: 0.480000; val_acc: 0.467000
(Iteration 1001 / 1225) loss: 1.611103
(Iteration 1101 / 1225) loss: 1.415286
(Iteration 1201 / 1225) loss: 1.534181
(Epoch 5 / 5) train acc: 0.480000; val_acc: 0.491000
(Iteration 1 / 815) loss: 2.365790
(Epoch 0 / 5) train acc: 0.103000; val_acc: 0.136000
(Iteration 101 / 815) loss: 1.940888
(Epoch 1 / 5) train acc: 0.335000; val_acc: 0.360000
(Iteration 201 / 815) loss: 1.824833
(Iteration 301 / 815) loss: 1.748234
(Epoch 2 / 5) train acc: 0.417000; val_acc: 0.403000
(Iteration 401 / 815) loss: 1.773234
(Epoch 3 / 5) train acc: 0.439000; val_acc: 0.433000
(Iteration 501 / 815) loss: 1.622081
(Iteration 601 / 815) loss: 1.521100
(Epoch 4 / 5) train acc: 0.476000; val_acc: 0.447000
(Iteration 701 / 815) loss: 1.612517
(Iteration 801 / 815) loss: 1.595560
(Epoch 5 / 5) train acc: 0.473000; val_acc: 0.459000
(Iteration 1 / 2450) loss: 2.384012
(Epoch 0 / 5) train acc: 0.121000; val_acc: 0.111000
(Iteration 101 / 2450) loss: 2.067359
(Iteration 201 / 2450) loss: 1.863463
(Iteration 301 / 2450) loss: 1.925608
(Iteration 401 / 2450) loss: 1.775563
(Epoch 1 / 5) train acc: 0.428000; val_acc: 0.424000
(Iteration 501 / 2450) loss: 1.810715
(Iteration 601 / 2450) loss: 1.598287
(Iteration 701 / 2450) loss: 1.575227
(Iteration 801 / 2450) loss: 1.746152
(Iteration 901 / 2450) loss: 1.715121
```

```
(Epoch 2 / 5) train acc: 0.431000; val_acc: 0.456000
(Iteration 1001 / 2450) loss: 1.486822
(Iteration 1101 / 2450) loss: 1.494195
(Iteration 1201 / 2450) loss: 1.345785
(Iteration 1301 / 2450) loss: 1.584626
(Iteration 1401 / 2450) loss: 1.549099
(Epoch 3 / 5) train acc: 0.504000; val_acc: 0.459000
(Iteration 1501 / 2450) loss: 1.560845
(Iteration 1601 / 2450) loss: 1.585783
(Iteration 1701 / 2450) loss: 1.590825
(Iteration 1801 / 2450) loss: 1.618187
(Iteration 1901 / 2450) loss: 1.417265
(Epoch 4 / 5) train acc: 0.517000; val_acc: 0.491000
(Iteration 2001 / 2450) loss: 1.397381
(Iteration 2101 / 2450) loss: 1.542422
(Iteration 2201 / 2450) loss: 1.497342
(Iteration 2301 / 2450) loss: 1.625837
(Iteration 2401 / 2450) loss: 1.543703
(Epoch 5 / 5) train acc: 0.544000; val_acc: 0.475000
(Iteration 1 / 1225) loss: 2.376382
(Epoch 0 / 5) train acc: 0.124000; val_acc: 0.120000
(Iteration 101 / 1225) loss: 1.958526
(Iteration 201 / 1225) loss: 1.890026
(Epoch 1 / 5) train acc: 0.377000; val_acc: 0.391000
(Iteration 301 / 1225) loss: 1.756481
(Iteration 401 / 1225) loss: 1.781075
(Epoch 2 / 5) train acc: 0.450000; val_acc: 0.434000
(Iteration 501 / 1225) loss: 1.672344
(Iteration 601 / 1225) loss: 1.662382
(Iteration 701 / 1225) loss: 1.587073
(Epoch 3 / 5) train acc: 0.463000; val_acc: 0.453000
(Iteration 801 / 1225) loss: 1.699036
(Iteration 901 / 1225) loss: 1.544397
(Epoch 4 / 5) train acc: 0.503000; val_acc: 0.460000
(Iteration 1001 / 1225) loss: 1.620147
(Iteration 1101 / 1225) loss: 1.655970
(Iteration 1201 / 1225) loss: 1.506378
(Epoch 5 / 5) train acc: 0.498000; val_acc: 0.465000
(Iteration 1 / 815) loss: 2.379150
(Epoch 0 / 5) train acc: 0.147000; val_acc: 0.144000
(Iteration 101 / 815) loss: 2.032309
(Epoch 1 / 5) train acc: 0.359000; val_acc: 0.353000
(Iteration 201 / 815) loss: 1.831806
(Iteration 301 / 815) loss: 1.690210
(Epoch 2 / 5) train acc: 0.391000; val_acc: 0.395000
(Iteration 401 / 815) loss: 1.639675
(Epoch 3 / 5) train acc: 0.453000; val_acc: 0.433000
(Iteration 501 / 815) loss: 1.643047
```

```
(Iteration 601 / 815) loss: 1.586321
(Epoch 4 / 5) train acc: 0.415000; val_acc: 0.464000
(Iteration 701 / 815) loss: 1.682054
(Iteration 801 / 815) loss: 1.586195
(Epoch 5 / 5) train acc: 0.449000; val_acc: 0.455000
(Iteration 1 / 2450) loss: 2.395297
(Epoch 0 / 5) train acc: 0.092000; val_acc: 0.108000
(Iteration 101 / 2450) loss: 2.206317
(Iteration 201 / 2450) loss: 1.931555
(Iteration 301 / 2450) loss: 1.673449
(Iteration 401 / 2450) loss: 1.844599
(Epoch 1 / 5) train acc: 0.414000; val_acc: 0.434000
(Iteration 501 / 2450) loss: 1.566491
(Iteration 601 / 2450) loss: 1.839385
(Iteration 701 / 2450) loss: 1.678862
(Iteration 801 / 2450) loss: 1.656272
(Iteration 901 / 2450) loss: 1.718609
(Epoch 2 / 5) train acc: 0.466000; val_acc: 0.470000
(Iteration 1001 / 2450) loss: 1.475147
(Iteration 1101 / 2450) loss: 1.528926
(Iteration 1201 / 2450) loss: 1.732779
(Iteration 1301 / 2450) loss: 1.524485
(Iteration 1401 / 2450) loss: 1.390337
(Epoch 3 / 5) train acc: 0.506000; val_acc: 0.472000
(Iteration 1501 / 2450) loss: 1.488167
(Iteration 1601 / 2450) loss: 1.607789
(Iteration 1701 / 2450) loss: 1.488479
(Iteration 1801 / 2450) loss: 1.441976
(Iteration 1901 / 2450) loss: 1.481887
(Epoch 4 / 5) train acc: 0.514000; val_acc: 0.478000
(Iteration 2001 / 2450) loss: 1.532689
(Iteration 2101 / 2450) loss: 1.621655
(Iteration 2201 / 2450) loss: 1.492931
(Iteration 2301 / 2450) loss: 1.357073
(Iteration 2401 / 2450) loss: 1.775724
(Epoch 5 / 5) train acc: 0.539000; val_acc: 0.511000
(Iteration 1 / 1225) loss: 2.398806
(Epoch 0 / 5) train acc: 0.094000; val_acc: 0.091000
(Iteration 101 / 1225) loss: 2.006412
(Iteration 201 / 1225) loss: 1.920746
(Epoch 1 / 5) train acc: 0.359000; val_acc: 0.389000
(Iteration 301 / 1225) loss: 1.792720
(Iteration 401 / 1225) loss: 1.817529
(Epoch 2 / 5) train acc: 0.435000; val_acc: 0.451000
(Iteration 501 / 1225) loss: 1.737853
(Iteration 601 / 1225) loss: 1.656694
(Iteration 701 / 1225) loss: 1.624679
(Epoch 3 / 5) train acc: 0.450000; val_acc: 0.471000
```

```
(Iteration 801 / 1225) loss: 1.568509
(Iteration 901 / 1225) loss: 1.372979
(Epoch 4 / 5) train acc: 0.462000; val_acc: 0.471000
(Iteration 1001 / 1225) loss: 1.571675
(Iteration 1101 / 1225) loss: 1.584605
(Iteration 1201 / 1225) loss: 1.631948
(Epoch 5 / 5) train acc: 0.496000; val_acc: 0.480000
(Iteration 1 / 815) loss: 2.396356
(Epoch 0 / 5) train acc: 0.114000; val_acc: 0.127000
(Iteration 101 / 815) loss: 2.008331
(Epoch 1 / 5) train acc: 0.362000; val_acc: 0.354000
(Iteration 201 / 815) loss: 1.907021
(Iteration 301 / 815) loss: 1.806096
(Epoch 2 / 5) train acc: 0.397000; val_acc: 0.419000
(Iteration 401 / 815) loss: 1.686331
(Epoch 3 / 5) train acc: 0.419000; val_acc: 0.439000
(Iteration 501 / 815) loss: 1.853012
(Iteration 601 / 815) loss: 1.666796
(Epoch 4 / 5) train acc: 0.461000; val_acc: 0.438000
(Iteration 701 / 815) loss: 1.672057
(Iteration 801 / 815) loss: 1.650572
(Epoch 5 / 5) train acc: 0.473000; val_acc: 0.448000
(Iteration 1 / 2940) loss: 2.343360
(Epoch 0 / 6) train acc: 0.143000; val_acc: 0.130000
(Iteration 101 / 2940) loss: 2.053345
(Iteration 201 / 2940) loss: 1.817841
(Iteration 301 / 2940) loss: 1.677710
(Iteration 401 / 2940) loss: 1.656466
(Epoch 1 / 6) train acc: 0.427000; val_acc: 0.437000
(Iteration 501 / 2940) loss: 1.848649
(Iteration 601 / 2940) loss: 1.614059
(Iteration 701 / 2940) loss: 1.554207
(Iteration 801 / 2940) loss: 1.578498
(Iteration 901 / 2940) loss: 1.578163
(Epoch 2 / 6) train acc: 0.467000; val_acc: 0.468000
(Iteration 1001 / 2940) loss: 1.512352
(Iteration 1101 / 2940) loss: 1.603089
(Iteration 1201 / 2940) loss: 1.499867
(Iteration 1301 / 2940) loss: 1.426550
(Iteration 1401 / 2940) loss: 1.466570
(Epoch 3 / 6) train acc: 0.500000; val_acc: 0.465000
(Iteration 1501 / 2940) loss: 1.551983
(Iteration 1601 / 2940) loss: 1.537302
(Iteration 1701 / 2940) loss: 1.315485
(Iteration 1801 / 2940) loss: 1.444175
(Iteration 1901 / 2940) loss: 1.404046
(Epoch 4 / 6) train acc: 0.520000; val_acc: 0.477000
(Iteration 2001 / 2940) loss: 1.380420
```

```
(Iteration 2101 / 2940) loss: 1.619745
(Iteration 2201 / 2940) loss: 1.535306
(Iteration 2301 / 2940) loss: 1.550834
(Iteration 2401 / 2940) loss: 1.284111
(Epoch 5 / 6) train acc: 0.551000; val_acc: 0.490000
(Iteration 2501 / 2940) loss: 1.292583
(Iteration 2601 / 2940) loss: 1.254005
(Iteration 2701 / 2940) loss: 1.535439
(Iteration 2801 / 2940) loss: 1.415339
(Iteration 2901 / 2940) loss: 1.406691
(Epoch 6 / 6) train acc: 0.560000; val_acc: 0.506000
(Iteration 1 / 1470) loss: 2.348364
(Epoch 0 / 6) train acc: 0.129000; val_acc: 0.100000
(Iteration 101 / 1470) loss: 2.086373
(Iteration 201 / 1470) loss: 1.797330
(Epoch 1 / 6) train acc: 0.377000; val_acc: 0.382000
(Iteration 301 / 1470) loss: 1.872216
(Iteration 401 / 1470) loss: 1.767997
(Epoch 2 / 6) train acc: 0.470000; val_acc: 0.436000
(Iteration 501 / 1470) loss: 1.512397
(Iteration 601 / 1470) loss: 1.541424
(Iteration 701 / 1470) loss: 1.559823
(Epoch 3 / 6) train acc: 0.477000; val_acc: 0.446000
(Iteration 801 / 1470) loss: 1.501605
(Iteration 901 / 1470) loss: 1.512578
(Epoch 4 / 6) train acc: 0.451000; val_acc: 0.467000
(Iteration 1001 / 1470) loss: 1.598772
(Iteration 1101 / 1470) loss: 1.497307
(Iteration 1201 / 1470) loss: 1.412374
(Epoch 5 / 6) train acc: 0.505000; val_acc: 0.478000
(Iteration 1301 / 1470) loss: 1.418863
(Iteration 1401 / 1470) loss: 1.362471
(Epoch 6 / 6) train acc: 0.497000; val_acc: 0.475000
(Iteration 1 / 978) loss: 2.344886
(Epoch 0 / 6) train acc: 0.132000; val_acc: 0.118000
(Iteration 101 / 978) loss: 2.003466
(Epoch 1 / 6) train acc: 0.352000; val_acc: 0.344000
(Iteration 201 / 978) loss: 1.878538
(Iteration 301 / 978) loss: 1.762396
(Epoch 2 / 6) train acc: 0.426000; val_acc: 0.399000
(Iteration 401 / 978) loss: 1.688905
(Epoch 3 / 6) train acc: 0.433000; val_acc: 0.442000
(Iteration 501 / 978) loss: 1.689665
(Iteration 601 / 978) loss: 1.691134
(Epoch 4 / 6) train acc: 0.453000; val_acc: 0.447000
(Iteration 701 / 978) loss: 1.596136
(Iteration 801 / 978) loss: 1.592033
(Epoch 5 / 6) train acc: 0.492000; val_acc: 0.466000
```

```
(Iteration 901 / 978) loss: 1.550541
(Epoch 6 / 6) train acc: 0.474000; val_acc: 0.481000
(Iteration 1 / 2940) loss: 2.360421
(Epoch 0 / 6) train acc: 0.118000; val_acc: 0.112000
(Iteration 101 / 2940) loss: 2.067232
(Iteration 201 / 2940) loss: 1.978286
(Iteration 301 / 2940) loss: 1.730705
(Iteration 401 / 2940) loss: 1.786039
(Epoch 1 / 6) train acc: 0.417000; val_acc: 0.451000
(Iteration 501 / 2940) loss: 1.728870
(Iteration 601 / 2940) loss: 1.440974
(Iteration 701 / 2940) loss: 1.661018
(Iteration 801 / 2940) loss: 1.673617
(Iteration 901 / 2940) loss: 1.675118
(Epoch 2 / 6) train acc: 0.471000; val_acc: 0.461000
(Iteration 1001 / 2940) loss: 1.519421
(Iteration 1101 / 2940) loss: 1.554459
(Iteration 1201 / 2940) loss: 1.456666
(Iteration 1301 / 2940) loss: 1.495485
(Iteration 1401 / 2940) loss: 1.539645
(Epoch 3 / 6) train acc: 0.486000; val_acc: 0.489000
(Iteration 1501 / 2940) loss: 1.359537
(Iteration 1601 / 2940) loss: 1.557141
(Iteration 1701 / 2940) loss: 1.505316
(Iteration 1801 / 2940) loss: 1.483068
(Iteration 1901 / 2940) loss: 1.522721
(Epoch 4 / 6) train acc: 0.491000; val_acc: 0.490000
(Iteration 2001 / 2940) loss: 1.346052
(Iteration 2101 / 2940) loss: 1.339140
(Iteration 2201 / 2940) loss: 1.756243
(Iteration 2301 / 2940) loss: 1.481681
(Iteration 2401 / 2940) loss: 1.318164
(Epoch 5 / 6) train acc: 0.505000; val_acc: 0.497000
(Iteration 2501 / 2940) loss: 1.488283
(Iteration 2601 / 2940) loss: 1.551007
(Iteration 2701 / 2940) loss: 1.519111
(Iteration 2801 / 2940) loss: 1.579135
(Iteration 2901 / 2940) loss: 1.370480
(Epoch 6 / 6) train acc: 0.530000; val_acc: 0.515000
(Iteration 1 / 1470) loss: 2.366241
(Epoch 0 / 6) train acc: 0.095000; val_acc: 0.096000
(Iteration 101 / 1470) loss: 1.950677
(Iteration 201 / 1470) loss: 1.804650
(Epoch 1 / 6) train acc: 0.388000; val_acc: 0.391000
(Iteration 301 / 1470) loss: 1.686059
(Iteration 401 / 1470) loss: 1.695500
(Epoch 2 / 6) train acc: 0.434000; val_acc: 0.445000
(Iteration 501 / 1470) loss: 1.567246
```

```
(Iteration 601 / 1470) loss: 1.615246
(Iteration 701 / 1470) loss: 1.633517
(Epoch 3 / 6) train acc: 0.453000; val_acc: 0.457000
(Iteration 801 / 1470) loss: 1.553762
(Iteration 901 / 1470) loss: 1.661359
(Epoch 4 / 6) train acc: 0.473000; val_acc: 0.472000
(Iteration 1001 / 1470) loss: 1.546368
(Iteration 1101 / 1470) loss: 1.521748
(Iteration 1201 / 1470) loss: 1.502220
(Epoch 5 / 6) train acc: 0.496000; val_acc: 0.474000
(Iteration 1301 / 1470) loss: 1.529603
(Iteration 1401 / 1470) loss: 1.499357
(Epoch 6 / 6) train acc: 0.534000; val_acc: 0.483000
(Iteration 1 / 978) loss: 2.364053
(Epoch 0 / 6) train acc: 0.105000; val_acc: 0.111000
(Iteration 101 / 978) loss: 1.997503
(Epoch 1 / 6) train acc: 0.354000; val_acc: 0.354000
(Iteration 201 / 978) loss: 1.847115
(Iteration 301 / 978) loss: 1.751811
(Epoch 2 / 6) train acc: 0.377000; val_acc: 0.416000
(Iteration 401 / 978) loss: 1.698209
(Epoch 3 / 6) train acc: 0.430000; val_acc: 0.437000
(Iteration 501 / 978) loss: 1.632883
(Iteration 601 / 978) loss: 1.711837
(Epoch 4 / 6) train acc: 0.468000; val_acc: 0.466000
(Iteration 701 / 978) loss: 1.650475
(Iteration 801 / 978) loss: 1.529660
(Epoch 5 / 6) train acc: 0.474000; val_acc: 0.457000
(Iteration 901 / 978) loss: 1.679307
(Epoch 6 / 6) train acc: 0.462000; val_acc: 0.474000
(Iteration 1 / 2940) loss: 2.383300
(Epoch 0 / 6) train acc: 0.109000; val_acc: 0.105000
(Iteration 101 / 2940) loss: 2.072157
(Iteration 201 / 2940) loss: 1.879154
(Iteration 301 / 2940) loss: 1.787611
(Iteration 401 / 2940) loss: 1.638285
(Epoch 1 / 6) train acc: 0.410000; val_acc: 0.444000
(Iteration 501 / 2940) loss: 1.493282
(Iteration 601 / 2940) loss: 1.657065
(Iteration 701 / 2940) loss: 1.740587
(Iteration 801 / 2940) loss: 1.593210
(Iteration 901 / 2940) loss: 1.554845
(Epoch 2 / 6) train acc: 0.484000; val_acc: 0.469000
(Iteration 1001 / 2940) loss: 1.368059
(Iteration 1101 / 2940) loss: 1.703033
(Iteration 1201 / 2940) loss: 1.729851
(Iteration 1301 / 2940) loss: 1.626711
(Iteration 1401 / 2940) loss: 1.621901
```

```
(Epoch 3 / 6) train acc: 0.467000; val_acc: 0.465000
(Iteration 1501 / 2940) loss: 1.694717
(Iteration 1601 / 2940) loss: 1.417550
(Iteration 1701 / 2940) loss: 1.542857
(Iteration 1801 / 2940) loss: 1.524613
(Iteration 1901 / 2940) loss: 1.316086
(Epoch 4 / 6) train acc: 0.537000; val_acc: 0.488000
(Iteration 2001 / 2940) loss: 1.367247
(Iteration 2101 / 2940) loss: 1.283671
(Iteration 2201 / 2940) loss: 1.409294
(Iteration 2301 / 2940) loss: 1.500576
(Iteration 2401 / 2940) loss: 1.469480
(Epoch 5 / 6) train acc: 0.517000; val_acc: 0.487000
(Iteration 2501 / 2940) loss: 1.267790
(Iteration 2601 / 2940) loss: 1.465361
(Iteration 2701 / 2940) loss: 1.505620
(Iteration 2801 / 2940) loss: 1.334277
(Iteration 2901 / 2940) loss: 1.105317
(Epoch 6 / 6) train acc: 0.557000; val_acc: 0.498000
(Iteration 1 / 1470) loss: 2.381627
(Epoch 0 / 6) train acc: 0.111000; val_acc: 0.104000
(Iteration 101 / 1470) loss: 2.068158
(Iteration 201 / 1470) loss: 1.987232
(Epoch 1 / 6) train acc: 0.386000; val_acc: 0.379000
(Iteration 301 / 1470) loss: 1.837980
(Iteration 401 / 1470) loss: 1.662914
(Epoch 2 / 6) train acc: 0.418000; val_acc: 0.441000
(Iteration 501 / 1470) loss: 1.535684
(Iteration 601 / 1470) loss: 1.746519
(Iteration 701 / 1470) loss: 1.654738
(Epoch 3 / 6) train acc: 0.481000; val_acc: 0.471000
(Iteration 801 / 1470) loss: 1.665029
(Iteration 901 / 1470) loss: 1.620489
(Epoch 4 / 6) train acc: 0.460000; val_acc: 0.468000
(Iteration 1001 / 1470) loss: 1.532350
(Iteration 1101 / 1470) loss: 1.643924
(Iteration 1201 / 1470) loss: 1.605302
(Epoch 5 / 6) train acc: 0.496000; val_acc: 0.485000
(Iteration 1301 / 1470) loss: 1.313073
(Iteration 1401 / 1470) loss: 1.634441
(Epoch 6 / 6) train acc: 0.506000; val_acc: 0.487000
(Iteration 1 / 978) loss: 2.378952
(Epoch 0 / 6) train acc: 0.122000; val_acc: 0.110000
(Iteration 101 / 978) loss: 2.081598
(Epoch 1 / 6) train acc: 0.373000; val_acc: 0.356000
(Iteration 201 / 978) loss: 1.835772
(Iteration 301 / 978) loss: 1.777182
(Epoch 2 / 6) train acc: 0.363000; val_acc: 0.412000
```

```
(Iteration 401 / 978) loss: 1.662946
(Epoch 3 / 6) train acc: 0.399000; val_acc: 0.441000
(Iteration 501 / 978) loss: 1.673057
(Iteration 601 / 978) loss: 1.656436
(Epoch 4 / 6) train acc: 0.462000; val_acc: 0.453000
(Iteration 701 / 978) loss: 1.581821
(Iteration 801 / 978) loss: 1.586311
(Epoch 5 / 6) train acc: 0.474000; val_acc: 0.465000
(Iteration 901 / 978) loss: 1.584752
(Epoch 6 / 6) train acc: 0.493000; val_acc: 0.475000
(Iteration 1 / 2940) loss: 2.394724
(Epoch 0 / 6) train acc: 0.138000; val_acc: 0.141000
(Iteration 101 / 2940) loss: 2.036035
(Iteration 201 / 2940) loss: 1.903730
(Iteration 301 / 2940) loss: 1.869699
(Iteration 401 / 2940) loss: 1.820126
(Epoch 1 / 6) train acc: 0.414000; val_acc: 0.416000
(Iteration 501 / 2940) loss: 1.775224
(Iteration 601 / 2940) loss: 1.708130
(Iteration 701 / 2940) loss: 1.774105
(Iteration 801 / 2940) loss: 1.748318
(Iteration 901 / 2940) loss: 1.715478
(Epoch 2 / 6) train acc: 0.467000; val_acc: 0.453000
(Iteration 1001 / 2940) loss: 1.465594
(Iteration 1101 / 2940) loss: 1.644712
(Iteration 1201 / 2940) loss: 1.648927
(Iteration 1301 / 2940) loss: 1.675236
(Iteration 1401 / 2940) loss: 1.405890
(Epoch 3 / 6) train acc: 0.499000; val_acc: 0.476000
(Iteration 1501 / 2940) loss: 1.608249
(Iteration 1601 / 2940) loss: 1.403108
(Iteration 1701 / 2940) loss: 1.558017
(Iteration 1801 / 2940) loss: 1.615401
(Iteration 1901 / 2940) loss: 1.480655
(Epoch 4 / 6) train acc: 0.510000; val_acc: 0.473000
(Iteration 2001 / 2940) loss: 1.628163
(Iteration 2101 / 2940) loss: 1.464448
(Iteration 2201 / 2940) loss: 1.558052
(Iteration 2301 / 2940) loss: 1.568141
(Iteration 2401 / 2940) loss: 1.649297
(Epoch 5 / 6) train acc: 0.504000; val_acc: 0.478000
(Iteration 2501 / 2940) loss: 1.487236
(Iteration 2601 / 2940) loss: 1.414356
(Iteration 2701 / 2940) loss: 1.492113
(Iteration 2801 / 2940) loss: 1.499641
(Iteration 2901 / 2940) loss: 1.555867
(Epoch 6 / 6) train acc: 0.550000; val_acc: 0.507000
(Iteration 1 / 1470) loss: 2.397110
```

```
(Epoch 0 / 6) train acc: 0.089000; val_acc: 0.099000
(Iteration 101 / 1470) loss: 1.962718
(Iteration 201 / 1470) loss: 1.921753
(Epoch 1 / 6) train acc: 0.386000; val_acc: 0.383000
(Iteration 301 / 1470) loss: 1.797209
(Iteration 401 / 1470) loss: 1.740515
(Epoch 2 / 6) train acc: 0.434000; val_acc: 0.446000
(Iteration 501 / 1470) loss: 1.642247
(Iteration 601 / 1470) loss: 1.622084
(Iteration 701 / 1470) loss: 1.717012
(Epoch 3 / 6) train acc: 0.458000; val_acc: 0.462000
(Iteration 801 / 1470) loss: 1.705697
(Iteration 901 / 1470) loss: 1.571680
(Epoch 4 / 6) train acc: 0.498000; val_acc: 0.456000
(Iteration 1001 / 1470) loss: 1.695743
(Iteration 1101 / 1470) loss: 1.529115
(Iteration 1201 / 1470) loss: 1.681336
(Epoch 5 / 6) train acc: 0.498000; val_acc: 0.468000
(Iteration 1301 / 1470) loss: 1.619457
(Iteration 1401 / 1470) loss: 1.402579
(Epoch 6 / 6) train acc: 0.501000; val_acc: 0.483000
(Iteration 1 / 978) loss: 2.393640
(Epoch 0 / 6) train acc: 0.145000; val_acc: 0.111000
(Iteration 101 / 978) loss: 1.994825
(Epoch 1 / 6) train acc: 0.360000; val_acc: 0.359000
(Iteration 201 / 978) loss: 1.867872
(Iteration 301 / 978) loss: 1.781516
(Epoch 2 / 6) train acc: 0.399000; val_acc: 0.405000
(Iteration 401 / 978) loss: 1.713048
(Epoch 3 / 6) train acc: 0.416000; val_acc: 0.446000
(Iteration 501 / 978) loss: 1.682271
(Iteration 601 / 978) loss: 1.652377
(Epoch 4 / 6) train acc: 0.459000; val_acc: 0.452000
(Iteration 701 / 978) loss: 1.687658
(Iteration 801 / 978) loss: 1.681310
(Epoch 5 / 6) train acc: 0.488000; val_acc: 0.473000
(Iteration 901 / 978) loss: 1.513511
(Epoch 6 / 6) train acc: 0.478000; val_acc: 0.476000
(Iteration 1 / 1960) loss: 2.383110
(Epoch 0 / 4) train acc: 0.149000; val_acc: 0.158000
(Iteration 101 / 1960) loss: 1.894558
(Iteration 201 / 1960) loss: 1.844353
(Iteration 301 / 1960) loss: 1.811520
(Iteration 401 / 1960) loss: 1.683762
(Epoch 1 / 4) train acc: 0.419000; val_acc: 0.450000
(Iteration 501 / 1960) loss: 1.649003
(Iteration 601 / 1960) loss: 1.651617
(Iteration 701 / 1960) loss: 1.589282
```

```
(Iteration 801 / 1960) loss: 1.638413
(Iteration 901 / 1960) loss: 1.579579
(Epoch 2 / 4) train acc: 0.488000; val_acc: 0.453000
(Iteration 1001 / 1960) loss: 1.562777
(Iteration 1101 / 1960) loss: 1.411285
(Iteration 1201 / 1960) loss: 1.667046
(Iteration 1301 / 1960) loss: 1.701408
(Iteration 1401 / 1960) loss: 1.555829
(Epoch 3 / 4) train acc: 0.509000; val_acc: 0.465000
(Iteration 1501 / 1960) loss: 1.503322
(Iteration 1601 / 1960) loss: 1.549613
(Iteration 1701 / 1960) loss: 1.481752
(Iteration 1801 / 1960) loss: 1.270734
(Iteration 1901 / 1960) loss: 1.549171
(Epoch 4 / 4) train acc: 0.531000; val_acc: 0.478000
(Iteration 1 / 980) loss: 2.397927
(Epoch 0 / 4) train acc: 0.133000; val_acc: 0.135000
(Iteration 101 / 980) loss: 2.033951
(Iteration 201 / 980) loss: 1.750662
(Epoch 1 / 4) train acc: 0.376000; val_acc: 0.417000
(Iteration 301 / 980) loss: 1.736154
(Iteration 401 / 980) loss: 1.575847
(Epoch 2 / 4) train acc: 0.446000; val_acc: 0.447000
(Iteration 501 / 980) loss: 1.712437
(Iteration 601 / 980) loss: 1.691135
(Iteration 701 / 980) loss: 1.592444
(Epoch 3 / 4) train acc: 0.456000; val_acc: 0.475000
(Iteration 801 / 980) loss: 1.627782
(Iteration 901 / 980) loss: 1.590045
(Epoch 4 / 4) train acc: 0.496000; val_acc: 0.477000
(Iteration 1 / 652) loss: 2.396300
(Epoch 0 / 4) train acc: 0.127000; val_acc: 0.121000
(Iteration 101 / 652) loss: 2.082586
(Epoch 1 / 4) train acc: 0.359000; val_acc: 0.364000
(Iteration 201 / 652) loss: 1.834166
(Iteration 301 / 652) loss: 1.759131
(Epoch 2 / 4) train acc: 0.425000; val_acc: 0.428000
(Iteration 401 / 652) loss: 1.690954
(Epoch 3 / 4) train acc: 0.427000; val_acc: 0.446000
(Iteration 501 / 652) loss: 1.675119
(Iteration 601 / 652) loss: 1.595201
(Epoch 4 / 4) train acc: 0.451000; val_acc: 0.464000
(Iteration 1 / 1960) loss: 2.429952
(Epoch 0 / 4) train acc: 0.096000; val_acc: 0.112000
(Iteration 101 / 1960) loss: 2.119322
(Iteration 201 / 1960) loss: 1.879395
(Iteration 301 / 1960) loss: 1.651041
(Iteration 401 / 1960) loss: 1.769338
```

```
(Epoch 1 / 4) train acc: 0.436000; val_acc: 0.436000
(Iteration 501 / 1960) loss: 1.944563
(Iteration 601 / 1960) loss: 1.727093
(Iteration 701 / 1960) loss: 1.610424
(Iteration 801 / 1960) loss: 1.519288
(Iteration 901 / 1960) loss: 1.632929
(Epoch 2 / 4) train acc: 0.498000; val_acc: 0.476000
(Iteration 1001 / 1960) loss: 1.832413
(Iteration 1101 / 1960) loss: 1.534793
(Iteration 1201 / 1960) loss: 1.498287
(Iteration 1301 / 1960) loss: 1.547885
(Iteration 1401 / 1960) loss: 1.517551
(Epoch 3 / 4) train acc: 0.504000; val_acc: 0.475000
(Iteration 1501 / 1960) loss: 1.411673
(Iteration 1601 / 1960) loss: 1.441994
(Iteration 1701 / 1960) loss: 1.676662
(Iteration 1801 / 1960) loss: 1.611314
(Iteration 1901 / 1960) loss: 1.287366
(Epoch 4 / 4) train acc: 0.484000; val_acc: 0.484000
(Iteration 1 / 980) loss: 2.426733
(Epoch 0 / 4) train acc: 0.130000; val_acc: 0.127000
(Iteration 101 / 980) loss: 2.038864
(Iteration 201 / 980) loss: 1.814598
(Epoch 1 / 4) train acc: 0.375000; val_acc: 0.418000
(Iteration 301 / 980) loss: 1.751228
(Iteration 401 / 980) loss: 1.689471
(Epoch 2 / 4) train acc: 0.446000; val_acc: 0.451000
(Iteration 501 / 980) loss: 1.672063
(Iteration 601 / 980) loss: 1.494263
(Iteration 701 / 980) loss: 1.717795
(Epoch 3 / 4) train acc: 0.497000; val_acc: 0.462000
(Iteration 801 / 980) loss: 1.656337
(Iteration 901 / 980) loss: 1.559571
(Epoch 4 / 4) train acc: 0.477000; val_acc: 0.483000
(Iteration 1 / 652) loss: 2.432346
(Epoch 0 / 4) train acc: 0.090000; val_acc: 0.089000
(Iteration 101 / 652) loss: 2.026108
(Epoch 1 / 4) train acc: 0.397000; val_acc: 0.371000
(Iteration 201 / 652) loss: 1.863849
(Iteration 301 / 652) loss: 1.808782
(Epoch 2 / 4) train acc: 0.419000; val_acc: 0.436000
(Iteration 401 / 652) loss: 1.789450
(Epoch 3 / 4) train acc: 0.456000; val_acc: 0.451000
(Iteration 501 / 652) loss: 1.655371
(Iteration 601 / 652) loss: 1.720560
(Epoch 4 / 4) train acc: 0.436000; val_acc: 0.469000
(Iteration 1 / 1960) loss: 2.462887
(Epoch 0 / 4) train acc: 0.118000; val_acc: 0.120000
```

```
(Iteration 101 / 1960) loss: 1.988818
(Iteration 201 / 1960) loss: 1.891778
(Iteration 301 / 1960) loss: 1.670526
(Iteration 401 / 1960) loss: 1.754258
(Epoch 1 / 4) train acc: 0.449000; val_acc: 0.458000
(Iteration 501 / 1960) loss: 1.695279
(Iteration 601 / 1960) loss: 1.750733
(Iteration 701 / 1960) loss: 1.714186
(Iteration 801 / 1960) loss: 1.636153
(Iteration 901 / 1960) loss: 1.667171
(Epoch 2 / 4) train acc: 0.471000; val_acc: 0.465000
(Iteration 1001 / 1960) loss: 1.809721
(Iteration 1101 / 1960) loss: 1.495466
(Iteration 1201 / 1960) loss: 1.515671
(Iteration 1301 / 1960) loss: 1.444106
(Iteration 1401 / 1960) loss: 1.564650
(Epoch 3 / 4) train acc: 0.515000; val_acc: 0.492000
(Iteration 1501 / 1960) loss: 1.456023
(Iteration 1601 / 1960) loss: 1.585300
(Iteration 1701 / 1960) loss: 1.420767
(Iteration 1801 / 1960) loss: 1.365512
(Iteration 1901 / 1960) loss: 1.566754
(Epoch 4 / 4) train acc: 0.535000; val_acc: 0.515000
(Iteration 1 / 980) loss: 2.457917
(Epoch 0 / 4) train acc: 0.128000; val_acc: 0.134000
(Iteration 101 / 980) loss: 2.056002
(Iteration 201 / 980) loss: 1.946599
(Epoch 1 / 4) train acc: 0.396000; val_acc: 0.412000
(Iteration 301 / 980) loss: 1.916563
(Iteration 401 / 980) loss: 1.712727
(Epoch 2 / 4) train acc: 0.455000; val_acc: 0.440000
(Iteration 501 / 980) loss: 1.713500
(Iteration 601 / 980) loss: 1.669414
(Iteration 701 / 980) loss: 1.689136
(Epoch 3 / 4) train acc: 0.461000; val_acc: 0.464000
(Iteration 801 / 980) loss: 1.631330
(Iteration 901 / 980) loss: 1.568235
(Epoch 4 / 4) train acc: 0.463000; val_acc: 0.477000
(Iteration 1 / 652) loss: 2.458848
(Epoch 0 / 4) train acc: 0.111000; val_acc: 0.123000
(Iteration 101 / 652) loss: 2.010480
(Epoch 1 / 4) train acc: 0.367000; val_acc: 0.369000
(Iteration 201 / 652) loss: 1.892555
(Iteration 301 / 652) loss: 1.797778
(Epoch 2 / 4) train acc: 0.423000; val_acc: 0.430000
(Iteration 401 / 652) loss: 1.760894
(Epoch 3 / 4) train acc: 0.432000; val_acc: 0.458000
(Iteration 501 / 652) loss: 1.759405
```

```
(Iteration 601 / 652) loss: 1.651233
(Epoch 4 / 4) train acc: 0.468000; val_acc: 0.458000
(Iteration 1 / 1960) loss: 2.488680
(Epoch 0 / 4) train acc: 0.119000; val_acc: 0.110000
(Iteration 101 / 1960) loss: 2.156573
(Iteration 201 / 1960) loss: 1.963272
(Iteration 301 / 1960) loss: 1.712629
(Iteration 401 / 1960) loss: 1.723627
(Epoch 1 / 4) train acc: 0.453000; val_acc: 0.452000
(Iteration 501 / 1960) loss: 1.706038
(Iteration 601 / 1960) loss: 1.716847
(Iteration 701 / 1960) loss: 1.720036
(Iteration 801 / 1960) loss: 1.839483
(Iteration 901 / 1960) loss: 1.646861
(Epoch 2 / 4) train acc: 0.467000; val_acc: 0.469000
(Iteration 1001 / 1960) loss: 1.543424
(Iteration 1101 / 1960) loss: 1.456428
(Iteration 1201 / 1960) loss: 1.697468
(Iteration 1301 / 1960) loss: 1.700695
(Iteration 1401 / 1960) loss: 1.501870
(Epoch 3 / 4) train acc: 0.501000; val_acc: 0.497000
(Iteration 1501 / 1960) loss: 1.544975
(Iteration 1601 / 1960) loss: 1.643433
(Iteration 1701 / 1960) loss: 1.502965
(Iteration 1801 / 1960) loss: 1.705489
(Iteration 1901 / 1960) loss: 1.536814
(Epoch 4 / 4) train acc: 0.531000; val_acc: 0.493000
(Iteration 1 / 980) loss: 2.487435
(Epoch 0 / 4) train acc: 0.111000; val_acc: 0.118000
(Iteration 101 / 980) loss: 2.075735
(Iteration 201 / 980) loss: 1.912334
(Epoch 1 / 4) train acc: 0.380000; val_acc: 0.397000
(Iteration 301 / 980) loss: 1.958226
(Iteration 401 / 980) loss: 1.701994
(Epoch 2 / 4) train acc: 0.462000; val_acc: 0.443000
(Iteration 501 / 980) loss: 1.745631
(Iteration 601 / 980) loss: 1.782283
(Iteration 701 / 980) loss: 1.737340
(Epoch 3 / 4) train acc: 0.470000; val_acc: 0.465000
(Iteration 801 / 980) loss: 1.719044
(Iteration 901 / 980) loss: 1.806965
(Epoch 4 / 4) train acc: 0.502000; val_acc: 0.470000
(Iteration 1 / 652) loss: 2.484703
(Epoch 0 / 4) train acc: 0.120000; val_acc: 0.120000
(Iteration 101 / 652) loss: 2.056333
(Epoch 1 / 4) train acc: 0.370000; val_acc: 0.378000
(Iteration 201 / 652) loss: 1.941910
(Iteration 301 / 652) loss: 1.780756
```

```
(Epoch 2 / 4) train acc: 0.421000; val_acc: 0.410000
(Iteration 401 / 652) loss: 1.807903
(Epoch 3 / 4) train acc: 0.449000; val_acc: 0.446000
(Iteration 501 / 652) loss: 1.750265
(Iteration 601 / 652) loss: 1.761102
(Epoch 4 / 4) train acc: 0.444000; val_acc: 0.459000
(Iteration 1 / 2450) loss: 2.404043
(Epoch 0 / 5) train acc: 0.104000; val_acc: 0.095000
(Iteration 101 / 2450) loss: 1.914623
(Iteration 201 / 2450) loss: 1.864976
(Iteration 301 / 2450) loss: 1.760890
(Iteration 401 / 2450) loss: 1.711415
(Epoch 1 / 5) train acc: 0.422000; val_acc: 0.444000
(Iteration 501 / 2450) loss: 1.761533
(Iteration 601 / 2450) loss: 1.549386
(Iteration 701 / 2450) loss: 1.588704
(Iteration 801 / 2450) loss: 1.663704
(Iteration 901 / 2450) loss: 1.714550
(Epoch 2 / 5) train acc: 0.484000; val_acc: 0.486000
(Iteration 1001 / 2450) loss: 1.666554
(Iteration 1101 / 2450) loss: 1.575240
(Iteration 1201 / 2450) loss: 1.649317
(Iteration 1301 / 2450) loss: 1.459380
(Iteration 1401 / 2450) loss: 1.443003
(Epoch 3 / 5) train acc: 0.540000; val_acc: 0.490000
(Iteration 1501 / 2450) loss: 1.628136
(Iteration 1601 / 2450) loss: 1.679279
(Iteration 1701 / 2450) loss: 1.512377
(Iteration 1801 / 2450) loss: 1.422236
(Iteration 1901 / 2450) loss: 1.388164
(Epoch 4 / 5) train acc: 0.530000; val_acc: 0.485000
(Iteration 2001 / 2450) loss: 1.378910
(Iteration 2101 / 2450) loss: 1.218953
(Iteration 2201 / 2450) loss: 1.366479
(Iteration 2301 / 2450) loss: 1.303874
(Iteration 2401 / 2450) loss: 1.424840
(Epoch 5 / 5) train acc: 0.530000; val_acc: 0.500000
(Iteration 1 / 1225) loss: 2.393582
(Epoch 0 / 5) train acc: 0.114000; val_acc: 0.105000
(Iteration 101 / 1225) loss: 2.012487
(Iteration 201 / 1225) loss: 1.876673
(Epoch 1 / 5) train acc: 0.395000; val_acc: 0.397000
(Iteration 301 / 1225) loss: 1.746972
(Iteration 401 / 1225) loss: 1.772579
(Epoch 2 / 5) train acc: 0.436000; val_acc: 0.451000
(Iteration 501 / 1225) loss: 1.671316
(Iteration 601 / 1225) loss: 1.657690
(Iteration 701 / 1225) loss: 1.599433
```

```
(Epoch 3 / 5) train acc: 0.479000; val_acc: 0.454000
(Iteration 801 / 1225) loss: 1.531698
(Iteration 901 / 1225) loss: 1.588220
(Epoch 4 / 5) train acc: 0.492000; val_acc: 0.465000
(Iteration 1001 / 1225) loss: 1.591601
(Iteration 1101 / 1225) loss: 1.508140
(Iteration 1201 / 1225) loss: 1.554805
(Epoch 5 / 5) train acc: 0.496000; val_acc: 0.476000
(Iteration 1 / 815) loss: 2.395259
(Epoch 0 / 5) train acc: 0.128000; val_acc: 0.120000
(Iteration 101 / 815) loss: 2.046396
(Epoch 1 / 5) train acc: 0.352000; val_acc: 0.371000
(Iteration 201 / 815) loss: 1.819561
(Iteration 301 / 815) loss: 1.795713
(Epoch 2 / 5) train acc: 0.445000; val_acc: 0.433000
(Iteration 401 / 815) loss: 1.684708
(Epoch 3 / 5) train acc: 0.451000; val_acc: 0.458000
(Iteration 501 / 815) loss: 1.625929
(Iteration 601 / 815) loss: 1.674034
(Epoch 4 / 5) train acc: 0.468000; val_acc: 0.466000
(Iteration 701 / 815) loss: 1.663935
(Iteration 801 / 815) loss: 1.591731
(Epoch 5 / 5) train acc: 0.473000; val_acc: 0.471000
(Iteration 1 / 2450) loss: 2.428368
(Epoch 0 / 5) train acc: 0.147000; val_acc: 0.138000
(Iteration 101 / 2450) loss: 2.020527
(Iteration 201 / 2450) loss: 2.008699
(Iteration 301 / 2450) loss: 1.952915
(Iteration 401 / 2450) loss: 1.797118
(Epoch 1 / 5) train acc: 0.441000; val_acc: 0.445000
(Iteration 501 / 2450) loss: 1.566333
(Iteration 601 / 2450) loss: 1.732322
(Iteration 701 / 2450) loss: 1.621949
(Iteration 801 / 2450) loss: 1.388686
(Iteration 901 / 2450) loss: 1.474288
(Epoch 2 / 5) train acc: 0.490000; val_acc: 0.463000
(Iteration 1001 / 2450) loss: 1.719046
(Iteration 1101 / 2450) loss: 1.488517
(Iteration 1201 / 2450) loss: 1.470184
(Iteration 1301 / 2450) loss: 1.586082
(Iteration 1401 / 2450) loss: 1.663272
(Epoch 3 / 5) train acc: 0.497000; val_acc: 0.482000
(Iteration 1501 / 2450) loss: 1.411733
(Iteration 1601 / 2450) loss: 1.343755
(Iteration 1701 / 2450) loss: 1.509544
(Iteration 1801 / 2450) loss: 1.477911
(Iteration 1901 / 2450) loss: 1.418641
(Epoch 4 / 5) train acc: 0.552000; val_acc: 0.492000
```

```
(Iteration 2001 / 2450) loss: 1.556201
(Iteration 2101 / 2450) loss: 1.434783
(Iteration 2201 / 2450) loss: 1.376738
(Iteration 2301 / 2450) loss: 1.552923
(Iteration 2401 / 2450) loss: 1.438750
(Epoch 5 / 5) train acc: 0.561000; val_acc: 0.517000
(Iteration 1 / 1225) loss: 2.424005
(Epoch 0 / 5) train acc: 0.135000; val_acc: 0.108000
(Iteration 101 / 1225) loss: 2.008345
(Iteration 201 / 1225) loss: 1.905552
(Epoch 1 / 5) train acc: 0.384000; val_acc: 0.404000
(Iteration 301 / 1225) loss: 1.803204
(Iteration 401 / 1225) loss: 1.787003
(Epoch 2 / 5) train acc: 0.455000; val_acc: 0.443000
(Iteration 501 / 1225) loss: 1.704668
(Iteration 601 / 1225) loss: 1.691630
(Iteration 701 / 1225) loss: 1.696003
(Epoch 3 / 5) train acc: 0.468000; val_acc: 0.475000
(Iteration 801 / 1225) loss: 1.591607
(Iteration 901 / 1225) loss: 1.528224
(Epoch 4 / 5) train acc: 0.486000; val_acc: 0.486000
(Iteration 1001 / 1225) loss: 1.557074
(Iteration 1101 / 1225) loss: 1.500705
(Iteration 1201 / 1225) loss: 1.473072
(Epoch 5 / 5) train acc: 0.482000; val_acc: 0.484000
(Iteration 1 / 815) loss: 2.427010
(Epoch 0 / 5) train acc: 0.132000; val_acc: 0.133000
(Iteration 101 / 815) loss: 2.012083
(Epoch 1 / 5) train acc: 0.385000; val_acc: 0.372000
(Iteration 201 / 815) loss: 1.930851
(Iteration 301 / 815) loss: 1.812487
(Epoch 2 / 5) train acc: 0.436000; val_acc: 0.430000
(Iteration 401 / 815) loss: 1.750379
(Epoch 3 / 5) train acc: 0.448000; val_acc: 0.454000
(Iteration 501 / 815) loss: 1.711063
(Iteration 601 / 815) loss: 1.726109
(Epoch 4 / 5) train acc: 0.450000; val_acc: 0.468000
(Iteration 701 / 815) loss: 1.649572
(Iteration 801 / 815) loss: 1.558338
(Epoch 5 / 5) train acc: 0.491000; val_acc: 0.460000
(Iteration 1 / 2450) loss: 2.454349
(Epoch 0 / 5) train acc: 0.125000; val_acc: 0.118000
(Iteration 101 / 2450) loss: 2.178431
(Iteration 201 / 2450) loss: 1.953292
(Iteration 301 / 2450) loss: 2.046882
(Iteration 401 / 2450) loss: 1.926070
(Epoch 1 / 5) train acc: 0.409000; val_acc: 0.428000
(Iteration 501 / 2450) loss: 1.674515
```

```
(Iteration 601 / 2450) loss: 1.653101
(Iteration 701 / 2450) loss: 1.883213
(Iteration 801 / 2450) loss: 1.495338
(Iteration 901 / 2450) loss: 1.749164
(Epoch 2 / 5) train acc: 0.463000; val_acc: 0.478000
(Iteration 1001 / 2450) loss: 1.471982
(Iteration 1101 / 2450) loss: 1.495553
(Iteration 1201 / 2450) loss: 1.589822
(Iteration 1301 / 2450) loss: 1.566635
(Iteration 1401 / 2450) loss: 1.530787
(Epoch 3 / 5) train acc: 0.487000; val_acc: 0.471000
(Iteration 1501 / 2450) loss: 1.459326
(Iteration 1601 / 2450) loss: 1.485513
(Iteration 1701 / 2450) loss: 1.476661
(Iteration 1801 / 2450) loss: 1.365881
(Iteration 1901 / 2450) loss: 1.543458
(Epoch 4 / 5) train acc: 0.541000; val_acc: 0.495000
(Iteration 2001 / 2450) loss: 1.533939
(Iteration 2101 / 2450) loss: 1.527750
(Iteration 2201 / 2450) loss: 1.328340
(Iteration 2301 / 2450) loss: 1.285101
(Iteration 2401 / 2450) loss: 1.516084
(Epoch 5 / 5) train acc: 0.569000; val_acc: 0.494000
(Iteration 1 / 1225) loss: 2.459462
(Epoch 0 / 5) train acc: 0.107000; val_acc: 0.095000
(Iteration 101 / 1225) loss: 2.075709
(Iteration 201 / 1225) loss: 1.831224
(Epoch 1 / 5) train acc: 0.402000; val_acc: 0.402000
(Iteration 301 / 1225) loss: 1.815877
(Iteration 401 / 1225) loss: 1.709913
(Epoch 2 / 5) train acc: 0.424000; val_acc: 0.448000
(Iteration 501 / 1225) loss: 1.840370
(Iteration 601 / 1225) loss: 1.798444
(Iteration 701 / 1225) loss: 1.673226
(Epoch 3 / 5) train acc: 0.473000; val_acc: 0.468000
(Iteration 801 / 1225) loss: 1.674897
(Iteration 901 / 1225) loss: 1.659746
(Epoch 4 / 5) train acc: 0.483000; val_acc: 0.476000
(Iteration 1001 / 1225) loss: 1.573161
(Iteration 1101 / 1225) loss: 1.465295
(Iteration 1201 / 1225) loss: 1.739799
(Epoch 5 / 5) train acc: 0.513000; val_acc: 0.480000
(Iteration 1 / 815) loss: 2.458585
(Epoch 0 / 5) train acc: 0.144000; val_acc: 0.121000
(Iteration 101 / 815) loss: 2.013180
(Epoch 1 / 5) train acc: 0.335000; val_acc: 0.369000
(Iteration 201 / 815) loss: 1.887068
(Iteration 301 / 815) loss: 1.868501
```

```
(Epoch 2 / 5) train acc: 0.376000; val_acc: 0.432000
(Iteration 401 / 815) loss: 1.793911
(Epoch 3 / 5) train acc: 0.445000; val_acc: 0.451000
(Iteration 501 / 815) loss: 1.793435
(Iteration 601 / 815) loss: 1.700770
(Epoch 4 / 5) train acc: 0.495000; val_acc: 0.455000
(Iteration 701 / 815) loss: 1.820591
(Iteration 801 / 815) loss: 1.559329
(Epoch 5 / 5) train acc: 0.457000; val_acc: 0.471000
(Iteration 1 / 2450) loss: 2.488368
(Epoch 0 / 5) train acc: 0.103000; val_acc: 0.084000
(Iteration 101 / 2450) loss: 2.196158
(Iteration 201 / 2450) loss: 1.988800
(Iteration 301 / 2450) loss: 1.905295
(Iteration 401 / 2450) loss: 1.629981
(Epoch 1 / 5) train acc: 0.429000; val_acc: 0.454000
(Iteration 501 / 2450) loss: 1.787431
(Iteration 601 / 2450) loss: 1.770002
(Iteration 701 / 2450) loss: 1.675993
(Iteration 801 / 2450) loss: 1.646043
(Iteration 901 / 2450) loss: 1.941804
(Epoch 2 / 5) train acc: 0.463000; val_acc: 0.457000
(Iteration 1001 / 2450) loss: 1.698728
(Iteration 1101 / 2450) loss: 1.775089
(Iteration 1201 / 2450) loss: 1.589929
(Iteration 1301 / 2450) loss: 1.457940
(Iteration 1401 / 2450) loss: 1.603185
(Epoch 3 / 5) train acc: 0.497000; val_acc: 0.466000
(Iteration 1501 / 2450) loss: 1.702829
(Iteration 1601 / 2450) loss: 1.540533
(Iteration 1701 / 2450) loss: 1.549288
(Iteration 1801 / 2450) loss: 1.542129
(Iteration 1901 / 2450) loss: 1.485328
(Epoch 4 / 5) train acc: 0.525000; val_acc: 0.483000
(Iteration 2001 / 2450) loss: 1.674557
(Iteration 2101 / 2450) loss: 1.428481
(Iteration 2201 / 2450) loss: 1.647064
(Iteration 2301 / 2450) loss: 1.479723
(Iteration 2401 / 2450) loss: 1.464492
(Epoch 5 / 5) train acc: 0.533000; val_acc: 0.499000
(Iteration 1 / 1225) loss: 2.485371
(Epoch 0 / 5) train acc: 0.089000; val_acc: 0.123000
(Iteration 101 / 1225) loss: 2.238408
(Iteration 201 / 1225) loss: 1.830258
(Epoch 1 / 5) train acc: 0.389000; val_acc: 0.406000
(Iteration 301 / 1225) loss: 1.937728
(Iteration 401 / 1225) loss: 1.812227
(Epoch 2 / 5) train acc: 0.426000; val_acc: 0.464000
```

```
(Iteration 501 / 1225) loss: 1.662318
(Iteration 601 / 1225) loss: 1.608664
(Iteration 701 / 1225) loss: 1.718441
(Epoch 3 / 5) train acc: 0.453000; val_acc: 0.469000
(Iteration 801 / 1225) loss: 1.706561
(Iteration 901 / 1225) loss: 1.579460
(Epoch 4 / 5) train acc: 0.514000; val_acc: 0.479000
(Iteration 1001 / 1225) loss: 1.518043
(Iteration 1101 / 1225) loss: 1.580132
(Iteration 1201 / 1225) loss: 1.512445
(Epoch 5 / 5) train acc: 0.524000; val_acc: 0.491000
(Iteration 1 / 815) loss: 2.488097
(Epoch 0 / 5) train acc: 0.101000; val_acc: 0.112000
(Iteration 101 / 815) loss: 2.009832
(Epoch 1 / 5) train acc: 0.360000; val_acc: 0.380000
(Iteration 201 / 815) loss: 1.922999
(Iteration 301 / 815) loss: 1.725821
(Epoch 2 / 5) train acc: 0.395000; val_acc: 0.424000
(Iteration 401 / 815) loss: 1.808533
(Epoch 3 / 5) train acc: 0.430000; val_acc: 0.443000
(Iteration 501 / 815) loss: 1.738763
(Iteration 601 / 815) loss: 1.595199
(Epoch 4 / 5) train acc: 0.490000; val_acc: 0.471000
(Iteration 701 / 815) loss: 1.720034
(Iteration 801 / 815) loss: 1.712519
(Epoch 5 / 5) train acc: 0.501000; val_acc: 0.481000
(Iteration 1 / 2940) loss: 2.392154
(Epoch 0 / 6) train acc: 0.116000; val_acc: 0.136000
(Iteration 101 / 2940) loss: 1.950403
(Iteration 201 / 2940) loss: 1.864012
(Iteration 301 / 2940) loss: 1.681761
(Iteration 401 / 2940) loss: 1.559496
(Epoch 1 / 6) train acc: 0.431000; val_acc: 0.439000
(Iteration 501 / 2940) loss: 1.673085
(Iteration 601 / 2940) loss: 1.613671
(Iteration 701 / 2940) loss: 1.749521
(Iteration 801 / 2940) loss: 1.934190
(Iteration 901 / 2940) loss: 1.624600
(Epoch 2 / 6) train acc: 0.477000; val_acc: 0.465000
(Iteration 1001 / 2940) loss: 1.516054
(Iteration 1101 / 2940) loss: 1.502105
(Iteration 1201 / 2940) loss: 1.421293
(Iteration 1301 / 2940) loss: 1.672934
(Iteration 1401 / 2940) loss: 1.377299
(Epoch 3 / 6) train acc: 0.499000; val_acc: 0.492000
(Iteration 1501 / 2940) loss: 1.621163
(Iteration 1601 / 2940) loss: 1.449770
(Iteration 1701 / 2940) loss: 1.508026
```

```
(Iteration 1801 / 2940) loss: 1.464415
(Iteration 1901 / 2940) loss: 1.602318
(Epoch 4 / 6) train acc: 0.544000; val_acc: 0.496000
(Iteration 2001 / 2940) loss: 1.525008
(Iteration 2101 / 2940) loss: 1.428437
(Iteration 2201 / 2940) loss: 1.388373
(Iteration 2301 / 2940) loss: 1.493569
(Iteration 2401 / 2940) loss: 1.165730
(Epoch 5 / 6) train acc: 0.554000; val_acc: 0.502000
(Iteration 2501 / 2940) loss: 1.247078
(Iteration 2601 / 2940) loss: 1.448462
(Iteration 2701 / 2940) loss: 1.427535
(Iteration 2801 / 2940) loss: 1.351432
(Iteration 2901 / 2940) loss: 1.377669
(Epoch 6 / 6) train acc: 0.523000; val_acc: 0.510000
(Iteration 1 / 1470) loss: 2.392597
(Epoch 0 / 6) train acc: 0.128000; val_acc: 0.177000
(Iteration 101 / 1470) loss: 1.944823
(Iteration 201 / 1470) loss: 1.880750
(Epoch 1 / 6) train acc: 0.404000; val_acc: 0.393000
(Iteration 301 / 1470) loss: 1.759773
(Iteration 401 / 1470) loss: 1.691163
(Epoch 2 / 6) train acc: 0.451000; val_acc: 0.453000
(Iteration 501 / 1470) loss: 1.621783
(Iteration 601 / 1470) loss: 1.521888
(Iteration 701 / 1470) loss: 1.689053
(Epoch 3 / 6) train acc: 0.477000; val_acc: 0.463000
(Iteration 801 / 1470) loss: 1.483959
(Iteration 901 / 1470) loss: 1.640227
(Epoch 4 / 6) train acc: 0.469000; val_acc: 0.466000
(Iteration 1001 / 1470) loss: 1.558014
(Iteration 1101 / 1470) loss: 1.492500
(Iteration 1201 / 1470) loss: 1.525434
(Epoch 5 / 6) train acc: 0.497000; val_acc: 0.482000
(Iteration 1301 / 1470) loss: 1.551047
(Iteration 1401 / 1470) loss: 1.419412
(Epoch 6 / 6) train acc: 0.507000; val_acc: 0.484000
(Iteration 1 / 978) loss: 2.395877
(Epoch 0 / 6) train acc: 0.121000; val_acc: 0.106000
(Iteration 101 / 978) loss: 1.996543
(Epoch 1 / 6) train acc: 0.365000; val_acc: 0.361000
(Iteration 201 / 978) loss: 1.869184
(Iteration 301 / 978) loss: 1.808058
(Epoch 2 / 6) train acc: 0.427000; val_acc: 0.431000
(Iteration 401 / 978) loss: 1.752617
(Epoch 3 / 6) train acc: 0.446000; val_acc: 0.449000
(Iteration 501 / 978) loss: 1.645760
(Iteration 601 / 978) loss: 1.637020
```

```
(Epoch 4 / 6) train acc: 0.470000; val_acc: 0.457000
(Iteration 701 / 978) loss: 1.624537
(Iteration 801 / 978) loss: 1.548510
(Epoch 5 / 6) train acc: 0.483000; val_acc: 0.472000
(Iteration 901 / 978) loss: 1.581999
(Epoch 6 / 6) train acc: 0.508000; val_acc: 0.469000
(Iteration 1 / 2940) loss: 2.428891
(Epoch 0 / 6) train acc: 0.101000; val_acc: 0.128000
(Iteration 101 / 2940) loss: 2.070321
(Iteration 201 / 2940) loss: 1.851683
(Iteration 301 / 2940) loss: 1.862841
(Iteration 401 / 2940) loss: 1.750568
(Epoch 1 / 6) train acc: 0.426000; val_acc: 0.454000
(Iteration 501 / 2940) loss: 1.826107
(Iteration 601 / 2940) loss: 1.812880
(Iteration 701 / 2940) loss: 1.625817
(Iteration 801 / 2940) loss: 1.712754
(Iteration 901 / 2940) loss: 1.540687
(Epoch 2 / 6) train acc: 0.468000; val_acc: 0.472000
(Iteration 1001 / 2940) loss: 1.549530
(Iteration 1101 / 2940) loss: 1.538641
(Iteration 1201 / 2940) loss: 1.576649
(Iteration 1301 / 2940) loss: 1.674919
(Iteration 1401 / 2940) loss: 1.746911
(Epoch 3 / 6) train acc: 0.483000; val_acc: 0.484000
(Iteration 1501 / 2940) loss: 1.481914
(Iteration 1601 / 2940) loss: 1.561037
(Iteration 1701 / 2940) loss: 1.498162
(Iteration 1801 / 2940) loss: 1.499948
(Iteration 1901 / 2940) loss: 1.557241
(Epoch 4 / 6) train acc: 0.543000; val_acc: 0.491000
(Iteration 2001 / 2940) loss: 1.810845
(Iteration 2101 / 2940) loss: 1.482835
(Iteration 2201 / 2940) loss: 1.594585
(Iteration 2301 / 2940) loss: 1.283370
(Iteration 2401 / 2940) loss: 1.313429
(Epoch 5 / 6) train acc: 0.544000; val_acc: 0.504000
(Iteration 2501 / 2940) loss: 1.343724
(Iteration 2601 / 2940) loss: 1.553131
(Iteration 2701 / 2940) loss: 1.518007
(Iteration 2801 / 2940) loss: 1.515537
(Iteration 2901 / 2940) loss: 1.317760
(Epoch 6 / 6) train acc: 0.553000; val_acc: 0.523000
(Iteration 1 / 1470) loss: 2.427238
(Epoch 0 / 6) train acc: 0.119000; val_acc: 0.145000
(Iteration 101 / 1470) loss: 2.045689
(Iteration 201 / 1470) loss: 1.859801
(Epoch 1 / 6) train acc: 0.376000; val_acc: 0.392000
```

```
(Iteration 301 / 1470) loss: 1.759594
(Iteration 401 / 1470) loss: 1.904278
(Epoch 2 / 6) train acc: 0.438000; val_acc: 0.455000
(Iteration 501 / 1470) loss: 1.817269
(Iteration 601 / 1470) loss: 1.618222
(Iteration 701 / 1470) loss: 1.596557
(Epoch 3 / 6) train acc: 0.512000; val_acc: 0.467000
(Iteration 801 / 1470) loss: 1.661018
(Iteration 901 / 1470) loss: 1.502623
(Epoch 4 / 6) train acc: 0.489000; val_acc: 0.481000
(Iteration 1001 / 1470) loss: 1.641546
(Iteration 1101 / 1470) loss: 1.497923
(Iteration 1201 / 1470) loss: 1.592645
(Epoch 5 / 6) train acc: 0.518000; val_acc: 0.486000
(Iteration 1301 / 1470) loss: 1.405612
(Iteration 1401 / 1470) loss: 1.401281
(Epoch 6 / 6) train acc: 0.515000; val_acc: 0.485000
(Iteration 1 / 978) loss: 2.430246
(Epoch 0 / 6) train acc: 0.106000; val_acc: 0.124000
(Iteration 101 / 978) loss: 1.995542
(Epoch 1 / 6) train acc: 0.346000; val_acc: 0.365000
(Iteration 201 / 978) loss: 1.816117
(Iteration 301 / 978) loss: 1.815599
(Epoch 2 / 6) train acc: 0.418000; val_acc: 0.429000
(Iteration 401 / 978) loss: 1.817856
(Epoch 3 / 6) train acc: 0.443000; val_acc: 0.453000
(Iteration 501 / 978) loss: 1.656243
(Iteration 601 / 978) loss: 1.604060
(Epoch 4 / 6) train acc: 0.473000; val_acc: 0.468000
(Iteration 701 / 978) loss: 1.687157
(Iteration 801 / 978) loss: 1.576498
(Epoch 5 / 6) train acc: 0.492000; val_acc: 0.468000
(Iteration 901 / 978) loss: 1.688245
(Epoch 6 / 6) train acc: 0.479000; val_acc: 0.481000
(Iteration 1 / 2940) loss: 2.454464
(Epoch 0 / 6) train acc: 0.149000; val_acc: 0.135000
(Iteration 101 / 2940) loss: 2.053420
(Iteration 201 / 2940) loss: 1.960073
(Iteration 301 / 2940) loss: 1.855089
(Iteration 401 / 2940) loss: 1.824138
(Epoch 1 / 6) train acc: 0.441000; val_acc: 0.432000
(Iteration 501 / 2940) loss: 1.627275
(Iteration 601 / 2940) loss: 1.655659
(Iteration 701 / 2940) loss: 1.573285
(Iteration 801 / 2940) loss: 1.781120
(Iteration 901 / 2940) loss: 1.600626
(Epoch 2 / 6) train acc: 0.523000; val_acc: 0.474000
(Iteration 1001 / 2940) loss: 1.531210
```

```
(Iteration 1101 / 2940) loss: 1.616168
(Iteration 1201 / 2940) loss: 1.578696
(Iteration 1301 / 2940) loss: 1.480350
(Iteration 1401 / 2940) loss: 1.585246
(Epoch 3 / 6) train acc: 0.490000; val_acc: 0.498000
(Iteration 1501 / 2940) loss: 1.576772
(Iteration 1601 / 2940) loss: 1.484899
(Iteration 1701 / 2940) loss: 1.852238
(Iteration 1801 / 2940) loss: 1.367874
(Iteration 1901 / 2940) loss: 1.595687
(Epoch 4 / 6) train acc: 0.497000; val_acc: 0.486000
(Iteration 2001 / 2940) loss: 1.547976
(Iteration 2101 / 2940) loss: 1.444281
(Iteration 2201 / 2940) loss: 1.507081
(Iteration 2301 / 2940) loss: 1.284874
(Iteration 2401 / 2940) loss: 1.659651
(Epoch 5 / 6) train acc: 0.521000; val_acc: 0.512000
(Iteration 2501 / 2940) loss: 1.528881
(Iteration 2601 / 2940) loss: 1.555244
(Iteration 2701 / 2940) loss: 1.404829
(Iteration 2801 / 2940) loss: 1.550575
(Iteration 2901 / 2940) loss: 1.443741
(Epoch 6 / 6) train acc: 0.550000; val_acc: 0.508000
(Iteration 1 / 1470) loss: 2.457577
(Epoch 0 / 6) train acc: 0.137000; val_acc: 0.116000
(Iteration 101 / 1470) loss: 2.061530
(Iteration 201 / 1470) loss: 1.839010
(Epoch 1 / 6) train acc: 0.404000; val_acc: 0.389000
(Iteration 301 / 1470) loss: 1.695647
(Iteration 401 / 1470) loss: 1.755940
(Epoch 2 / 6) train acc: 0.469000; val_acc: 0.455000
(Iteration 501 / 1470) loss: 1.674357
(Iteration 601 / 1470) loss: 1.681773
(Iteration 701 / 1470) loss: 1.754601
(Epoch 3 / 6) train acc: 0.489000; val_acc: 0.456000
(Iteration 801 / 1470) loss: 1.692678
(Iteration 901 / 1470) loss: 1.777253
(Epoch 4 / 6) train acc: 0.503000; val_acc: 0.470000
(Iteration 1001 / 1470) loss: 1.641415
(Iteration 1101 / 1470) loss: 1.698722
(Iteration 1201 / 1470) loss: 1.438359
(Epoch 5 / 6) train acc: 0.499000; val_acc: 0.482000
(Iteration 1301 / 1470) loss: 1.537172
(Iteration 1401 / 1470) loss: 1.463810
(Epoch 6 / 6) train acc: 0.516000; val_acc: 0.480000
(Iteration 1 / 978) loss: 2.461869
(Epoch 0 / 6) train acc: 0.112000; val_acc: 0.120000
(Iteration 101 / 978) loss: 1.985976
```

```
(Epoch 1 / 6) train acc: 0.354000; val_acc: 0.366000
(Iteration 201 / 978) loss: 1.881061
(Iteration 301 / 978) loss: 1.810248
(Epoch 2 / 6) train acc: 0.401000; val_acc: 0.438000
(Iteration 401 / 978) loss: 1.686782
(Epoch 3 / 6) train acc: 0.453000; val_acc: 0.450000
(Iteration 501 / 978) loss: 1.638454
(Iteration 601 / 978) loss: 1.678609
(Epoch 4 / 6) train acc: 0.432000; val_acc: 0.466000
(Iteration 701 / 978) loss: 1.613151
(Iteration 801 / 978) loss: 1.661183
(Epoch 5 / 6) train acc: 0.449000; val_acc: 0.460000
(Iteration 901 / 978) loss: 1.601423
(Epoch 6 / 6) train acc: 0.486000; val_acc: 0.472000
(Iteration 1 / 2940) loss: 2.489944
(Epoch 0 / 6) train acc: 0.106000; val_acc: 0.101000
(Iteration 101 / 2940) loss: 2.194715
(Iteration 201 / 2940) loss: 1.964494
(Iteration 301 / 2940) loss: 1.901940
(Iteration 401 / 2940) loss: 1.761959
(Epoch 1 / 6) train acc: 0.456000; val_acc: 0.453000
(Iteration 501 / 2940) loss: 1.811671
(Iteration 601 / 2940) loss: 1.672633
(Iteration 701 / 2940) loss: 1.680999
(Iteration 801 / 2940) loss: 1.653346
(Iteration 901 / 2940) loss: 1.629740
(Epoch 2 / 6) train acc: 0.487000; val_acc: 0.479000
(Iteration 1001 / 2940) loss: 1.656829
(Iteration 1101 / 2940) loss: 1.733212
(Iteration 1201 / 2940) loss: 1.576965
(Iteration 1301 / 2940) loss: 1.601695
(Iteration 1401 / 2940) loss: 1.448977
(Epoch 3 / 6) train acc: 0.509000; val_acc: 0.477000
(Iteration 1501 / 2940) loss: 1.762191
(Iteration 1601 / 2940) loss: 1.509796
(Iteration 1701 / 2940) loss: 1.652126
(Iteration 1801 / 2940) loss: 1.320611
(Iteration 1901 / 2940) loss: 1.568880
(Epoch 4 / 6) train acc: 0.553000; val_acc: 0.505000
(Iteration 2001 / 2940) loss: 1.464784
(Iteration 2101 / 2940) loss: 1.555676
(Iteration 2201 / 2940) loss: 1.407894
(Iteration 2301 / 2940) loss: 1.402475
(Iteration 2401 / 2940) loss: 1.523870
(Epoch 5 / 6) train acc: 0.542000; val_acc: 0.516000
(Iteration 2501 / 2940) loss: 1.481429
(Iteration 2601 / 2940) loss: 1.485371
(Iteration 2701 / 2940) loss: 1.365529
```

```
(Iteration 2801 / 2940) loss: 1.490259
(Iteration 2901 / 2940) loss: 1.419501
(Epoch 6 / 6) train acc: 0.555000; val_acc: 0.503000
(Iteration 1 / 1470) loss: 2.489627
(Epoch 0 / 6) train acc: 0.110000; val_acc: 0.106000
(Iteration 101 / 1470) loss: 2.103660
(Iteration 201 / 1470) loss: 1.979201
(Epoch 1 / 6) train acc: 0.410000; val_acc: 0.410000
(Iteration 301 / 1470) loss: 1.929645
(Iteration 401 / 1470) loss: 1.776821
(Epoch 2 / 6) train acc: 0.417000; val_acc: 0.430000
(Iteration 501 / 1470) loss: 1.707021
(Iteration 601 / 1470) loss: 1.710930
(Iteration 701 / 1470) loss: 1.720538
(Epoch 3 / 6) train acc: 0.481000; val_acc: 0.473000
(Iteration 801 / 1470) loss: 1.543473
(Iteration 901 / 1470) loss: 1.622821
(Epoch 4 / 6) train acc: 0.510000; val_acc: 0.484000
(Iteration 1001 / 1470) loss: 1.632184
(Iteration 1101 / 1470) loss: 1.701199
(Iteration 1201 / 1470) loss: 1.530887
(Epoch 5 / 6) train acc: 0.507000; val_acc: 0.489000
(Iteration 1301 / 1470) loss: 1.496321
(Iteration 1401 / 1470) loss: 1.777334
(Epoch 6 / 6) train acc: 0.495000; val_acc: 0.487000
(Iteration 1 / 978) loss: 2.488311
(Epoch 0 / 6) train acc: 0.143000; val_acc: 0.131000
(Iteration 101 / 978) loss: 2.049705
(Epoch 1 / 6) train acc: 0.344000; val_acc: 0.364000
(Iteration 201 / 978) loss: 1.968479
(Iteration 301 / 978) loss: 1.777721
(Epoch 2 / 6) train acc: 0.407000; val_acc: 0.424000
(Iteration 401 / 978) loss: 1.831042
(Epoch 3 / 6) train acc: 0.402000; val_acc: 0.460000
(Iteration 501 / 978) loss: 1.744028
(Iteration 601 / 978) loss: 1.759128
(Epoch 4 / 6) train acc: 0.451000; val_acc: 0.470000
(Iteration 701 / 978) loss: 1.626425
(Iteration 801 / 978) loss: 1.719066
(Epoch 5 / 6) train acc: 0.472000; val_acc: 0.484000
(Iteration 901 / 978) loss: 1.650815
(Epoch 6 / 6) train acc: 0.467000; val_acc: 0.479000
(Iteration 1 / 1960) loss: 2.438722
(Epoch 0 / 4) train acc: 0.108000; val_acc: 0.133000
(Iteration 101 / 1960) loss: 1.999466
(Iteration 201 / 1960) loss: 1.869471
(Iteration 301 / 1960) loss: 1.644318
(Iteration 401 / 1960) loss: 1.892500
```

```
(Epoch 1 / 4) train acc: 0.435000; val_acc: 0.435000
(Iteration 501 / 1960) loss: 1.939907
(Iteration 601 / 1960) loss: 1.739075
(Iteration 701 / 1960) loss: 1.535350
(Iteration 801 / 1960) loss: 1.616902
(Iteration 901 / 1960) loss: 1.506112
(Epoch 2 / 4) train acc: 0.479000; val_acc: 0.470000
(Iteration 1001 / 1960) loss: 1.510352
(Iteration 1101 / 1960) loss: 1.503719
(Iteration 1201 / 1960) loss: 1.531537
(Iteration 1301 / 1960) loss: 1.479810
(Iteration 1401 / 1960) loss: 1.445656
(Epoch 3 / 4) train acc: 0.513000; val_acc: 0.478000
(Iteration 1501 / 1960) loss: 1.378750
(Iteration 1601 / 1960) loss: 1.495370
(Iteration 1701 / 1960) loss: 1.547912
(Iteration 1801 / 1960) loss: 1.217067
(Iteration 1901 / 1960) loss: 1.427106
(Epoch 4 / 4) train acc: 0.511000; val_acc: 0.496000
(Iteration 1 / 980) loss: 2.436192
(Epoch 0 / 4) train acc: 0.150000; val_acc: 0.131000
(Iteration 101 / 980) loss: 1.956178
(Iteration 201 / 980) loss: 1.714653
(Epoch 1 / 4) train acc: 0.401000; val_acc: 0.420000
(Iteration 301 / 980) loss: 1.700343
(Iteration 401 / 980) loss: 1.662208
(Epoch 2 / 4) train acc: 0.458000; val_acc: 0.459000
(Iteration 501 / 980) loss: 1.763988
(Iteration 601 / 980) loss: 1.804447
(Iteration 701 / 980) loss: 1.719682
(Epoch 3 / 4) train acc: 0.476000; val_acc: 0.468000
(Iteration 801 / 980) loss: 1.666679
(Iteration 901 / 980) loss: 1.629887
(Epoch 4 / 4) train acc: 0.522000; val_acc: 0.478000
(Iteration 1 / 652) loss: 2.441553
(Epoch 0 / 4) train acc: 0.147000; val_acc: 0.155000
(Iteration 101 / 652) loss: 1.962918
(Epoch 1 / 4) train acc: 0.387000; val_acc: 0.376000
(Iteration 201 / 652) loss: 1.918608
(Iteration 301 / 652) loss: 1.811242
(Epoch 2 / 4) train acc: 0.437000; val_acc: 0.425000
(Iteration 401 / 652) loss: 1.679517
(Epoch 3 / 4) train acc: 0.431000; val_acc: 0.448000
(Iteration 501 / 652) loss: 1.738973
(Iteration 601 / 652) loss: 1.561069
(Epoch 4 / 4) train acc: 0.454000; val_acc: 0.474000
(Iteration 1 / 1960) loss: 2.489511
(Epoch 0 / 4) train acc: 0.101000; val_acc: 0.112000
```

```
(Iteration 101 / 1960) loss: 2.095451
(Iteration 201 / 1960) loss: 1.951480
(Iteration 301 / 1960) loss: 1.903347
(Iteration 401 / 1960) loss: 1.917223
(Epoch 1 / 4) train acc: 0.448000; val_acc: 0.463000
(Iteration 501 / 1960) loss: 1.842839
(Iteration 601 / 1960) loss: 1.890747
(Iteration 701 / 1960) loss: 1.640345
(Iteration 801 / 1960) loss: 1.665791
(Iteration 901 / 1960) loss: 1.634777
(Epoch 2 / 4) train acc: 0.498000; val_acc: 0.478000
(Iteration 1001 / 1960) loss: 1.756491
(Iteration 1101 / 1960) loss: 1.574351
(Iteration 1201 / 1960) loss: 1.581174
(Iteration 1301 / 1960) loss: 1.613362
(Iteration 1401 / 1960) loss: 1.688092
(Epoch 3 / 4) train acc: 0.526000; val_acc: 0.492000
(Iteration 1501 / 1960) loss: 1.529479
(Iteration 1601 / 1960) loss: 1.601393
(Iteration 1701 / 1960) loss: 1.542486
(Iteration 1801 / 1960) loss: 1.461550
(Iteration 1901 / 1960) loss: 1.462339
(Epoch 4 / 4) train acc: 0.533000; val_acc: 0.485000
(Iteration 1 / 980) loss: 2.487588
(Epoch 0 / 4) train acc: 0.134000; val_acc: 0.169000
(Iteration 101 / 980) loss: 2.020322
(Iteration 201 / 980) loss: 1.911545
(Epoch 1 / 4) train acc: 0.398000; val_acc: 0.389000
(Iteration 301 / 980) loss: 1.825638
(Iteration 401 / 980) loss: 1.724041
(Epoch 2 / 4) train acc: 0.439000; val_acc: 0.452000
(Iteration 501 / 980) loss: 1.740723
(Iteration 601 / 980) loss: 1.724159
(Iteration 701 / 980) loss: 1.646295
(Epoch 3 / 4) train acc: 0.476000; val_acc: 0.484000
(Iteration 801 / 980) loss: 1.555322
(Iteration 901 / 980) loss: 1.688729
(Epoch 4 / 4) train acc: 0.513000; val_acc: 0.483000
(Iteration 1 / 652) loss: 2.491085
(Epoch 0 / 4) train acc: 0.133000; val_acc: 0.119000
(Iteration 101 / 652) loss: 1.992812
(Epoch 1 / 4) train acc: 0.397000; val_acc: 0.380000
(Iteration 201 / 652) loss: 1.924486
(Iteration 301 / 652) loss: 1.833916
(Epoch 2 / 4) train acc: 0.395000; val_acc: 0.445000
(Iteration 401 / 652) loss: 1.724997
(Epoch 3 / 4) train acc: 0.455000; val_acc: 0.461000
(Iteration 501 / 652) loss: 1.846915
```

```
(Iteration 601 / 652) loss: 1.791944
(Epoch 4 / 4) train acc: 0.462000; val_acc: 0.468000
(Iteration 1 / 1960) loss: 2.539041
(Epoch 0 / 4) train acc: 0.134000; val_acc: 0.157000
(Iteration 101 / 1960) loss: 2.122482
(Iteration 201 / 1960) loss: 2.004141
(Iteration 301 / 1960) loss: 1.903639
(Iteration 401 / 1960) loss: 1.917512
(Epoch 1 / 4) train acc: 0.471000; val_acc: 0.448000
(Iteration 501 / 1960) loss: 1.753348
(Iteration 601 / 1960) loss: 1.717221
(Iteration 701 / 1960) loss: 1.701677
(Iteration 801 / 1960) loss: 1.634667
(Iteration 901 / 1960) loss: 1.601754
(Epoch 2 / 4) train acc: 0.456000; val_acc: 0.452000
(Iteration 1001 / 1960) loss: 1.937842
(Iteration 1101 / 1960) loss: 1.582422
(Iteration 1201 / 1960) loss: 1.470339
(Iteration 1301 / 1960) loss: 1.442262
(Iteration 1401 / 1960) loss: 1.606686
(Epoch 3 / 4) train acc: 0.502000; val_acc: 0.491000
(Iteration 1501 / 1960) loss: 1.493464
(Iteration 1601 / 1960) loss: 1.475050
(Iteration 1701 / 1960) loss: 1.476345
(Iteration 1801 / 1960) loss: 1.517553
(Iteration 1901 / 1960) loss: 1.552641
(Epoch 4 / 4) train acc: 0.532000; val_acc: 0.497000
(Iteration 1 / 980) loss: 2.532230
(Epoch 0 / 4) train acc: 0.117000; val_acc: 0.123000
(Iteration 101 / 980) loss: 2.071388
(Iteration 201 / 980) loss: 2.039719
(Epoch 1 / 4) train acc: 0.392000; val_acc: 0.403000
(Iteration 301 / 980) loss: 1.812829
(Iteration 401 / 980) loss: 1.866119
(Epoch 2 / 4) train acc: 0.445000; val_acc: 0.460000
(Iteration 501 / 980) loss: 1.671250
(Iteration 601 / 980) loss: 1.763309
(Iteration 701 / 980) loss: 1.738463
(Epoch 3 / 4) train acc: 0.489000; val_acc: 0.452000
(Iteration 801 / 980) loss: 1.533391
(Iteration 901 / 980) loss: 1.560349
(Epoch 4 / 4) train acc: 0.473000; val_acc: 0.476000
(Iteration 1 / 652) loss: 2.537696
(Epoch 0 / 4) train acc: 0.115000; val_acc: 0.098000
(Iteration 101 / 652) loss: 2.139873
(Epoch 1 / 4) train acc: 0.396000; val_acc: 0.391000
(Iteration 201 / 652) loss: 1.861157
(Iteration 301 / 652) loss: 1.934499
```

```
(Epoch 2 / 4) train acc: 0.412000; val_acc: 0.425000
(Iteration 401 / 652) loss: 1.805415
(Epoch 3 / 4) train acc: 0.469000; val_acc: 0.445000
(Iteration 501 / 652) loss: 1.817389
(Iteration 601 / 652) loss: 1.679248
(Epoch 4 / 4) train acc: 0.449000; val_acc: 0.461000
(Iteration 1 / 1960) loss: 2.595412
(Epoch 0 / 4) train acc: 0.107000; val_acc: 0.114000
(Iteration 101 / 1960) loss: 2.141167
(Iteration 201 / 1960) loss: 2.094783
(Iteration 301 / 1960) loss: 2.030794
(Iteration 401 / 1960) loss: 2.067204
(Epoch 1 / 4) train acc: 0.444000; val_acc: 0.439000
(Iteration 501 / 1960) loss: 1.753302
(Iteration 601 / 1960) loss: 1.738519
(Iteration 701 / 1960) loss: 1.769771
(Iteration 801 / 1960) loss: 1.744020
(Iteration 901 / 1960) loss: 1.742943
(Epoch 2 / 4) train acc: 0.464000; val_acc: 0.466000
(Iteration 1001 / 1960) loss: 1.605559
(Iteration 1101 / 1960) loss: 1.829985
(Iteration 1201 / 1960) loss: 1.699222
(Iteration 1301 / 1960) loss: 1.621810
(Iteration 1401 / 1960) loss: 1.741285
(Epoch 3 / 4) train acc: 0.475000; val_acc: 0.482000
(Iteration 1501 / 1960) loss: 1.666802
(Iteration 1601 / 1960) loss: 1.705312
(Iteration 1701 / 1960) loss: 1.556095
(Iteration 1801 / 1960) loss: 1.583723
(Iteration 1901 / 1960) loss: 1.507704
(Epoch 4 / 4) train acc: 0.552000; val_acc: 0.501000
(Iteration 1 / 980) loss: 2.586371
(Epoch 0 / 4) train acc: 0.112000; val_acc: 0.101000
(Iteration 101 / 980) loss: 2.138535
(Iteration 201 / 980) loss: 1.929113
(Epoch 1 / 4) train acc: 0.401000; val_acc: 0.411000
(Iteration 301 / 980) loss: 1.865612
(Iteration 401 / 980) loss: 1.758770
(Epoch 2 / 4) train acc: 0.425000; val_acc: 0.450000
(Iteration 501 / 980) loss: 1.766637
(Iteration 601 / 980) loss: 1.858542
(Iteration 701 / 980) loss: 1.687657
(Epoch 3 / 4) train acc: 0.480000; val_acc: 0.465000
(Iteration 801 / 980) loss: 1.784214
(Iteration 901 / 980) loss: 1.715449
(Epoch 4 / 4) train acc: 0.512000; val_acc: 0.463000
(Iteration 1 / 652) loss: 2.580203
(Epoch 0 / 4) train acc: 0.164000; val_acc: 0.162000
```

```
(Iteration 101 / 652) loss: 2.172260
(Epoch 1 / 4) train acc: 0.386000; val_acc: 0.367000
(Iteration 201 / 652) loss: 1.936154
(Iteration 301 / 652) loss: 1.876871
(Epoch 2 / 4) train acc: 0.402000; val_acc: 0.445000
(Iteration 401 / 652) loss: 1.789883
(Epoch 3 / 4) train acc: 0.466000; val_acc: 0.461000
(Iteration 501 / 652) loss: 1.850964
(Iteration 601 / 652) loss: 1.677163
(Epoch 4 / 4) train acc: 0.467000; val_acc: 0.470000
(Iteration 1 / 2450) loss: 2.439099
(Epoch 0 / 5) train acc: 0.132000; val_acc: 0.136000
(Iteration 101 / 2450) loss: 2.089477
(Iteration 201 / 2450) loss: 1.713409
(Iteration 301 / 2450) loss: 1.919581
(Iteration 401 / 2450) loss: 1.738794
(Epoch 1 / 5) train acc: 0.445000; val_acc: 0.445000
(Iteration 501 / 2450) loss: 1.669837
(Iteration 601 / 2450) loss: 1.642644
(Iteration 701 / 2450) loss: 1.612027
(Iteration 801 / 2450) loss: 1.763257
(Iteration 901 / 2450) loss: 1.686885
(Epoch 2 / 5) train acc: 0.508000; val_acc: 0.475000
(Iteration 1001 / 2450) loss: 1.633204
(Iteration 1101 / 2450) loss: 1.760081
(Iteration 1201 / 2450) loss: 1.481587
(Iteration 1301 / 2450) loss: 1.664659
(Iteration 1401 / 2450) loss: 1.654745
(Epoch 3 / 5) train acc: 0.500000; val_acc: 0.489000
(Iteration 1501 / 2450) loss: 1.473955
(Iteration 1601 / 2450) loss: 1.613833
(Iteration 1701 / 2450) loss: 1.579692
(Iteration 1801 / 2450) loss: 1.574689
(Iteration 1901 / 2450) loss: 1.437194
(Epoch 4 / 5) train acc: 0.524000; val_acc: 0.501000
(Iteration 2001 / 2450) loss: 1.491885
(Iteration 2101 / 2450) loss: 1.508550
(Iteration 2201 / 2450) loss: 1.452086
(Iteration 2301 / 2450) loss: 1.505661
(Iteration 2401 / 2450) loss: 1.481491
(Epoch 5 / 5) train acc: 0.550000; val_acc: 0.510000
(Iteration 1 / 1225) loss: 2.451085
(Epoch 0 / 5) train acc: 0.123000; val_acc: 0.117000
(Iteration 101 / 1225) loss: 2.029928
(Iteration 201 / 1225) loss: 1.875459
(Epoch 1 / 5) train acc: 0.405000; val_acc: 0.395000
(Iteration 301 / 1225) loss: 1.817831
(Iteration 401 / 1225) loss: 1.738616
```

```
(Epoch 2 / 5) train acc: 0.432000; val_acc: 0.463000
(Iteration 501 / 1225) loss: 1.591805
(Iteration 601 / 1225) loss: 1.782416
(Iteration 701 / 1225) loss: 1.575929
(Epoch 3 / 5) train acc: 0.459000; val_acc: 0.460000
(Iteration 801 / 1225) loss: 1.637204
(Iteration 901 / 1225) loss: 1.477936
(Epoch 4 / 5) train acc: 0.496000; val_acc: 0.486000
(Iteration 1001 / 1225) loss: 1.596321
(Iteration 1101 / 1225) loss: 1.523492
(Iteration 1201 / 1225) loss: 1.486601
(Epoch 5 / 5) train acc: 0.512000; val_acc: 0.507000
(Iteration 1 / 815) loss: 2.436413
(Epoch 0 / 5) train acc: 0.131000; val_acc: 0.129000
(Iteration 101 / 815) loss: 2.015144
(Epoch 1 / 5) train acc: 0.355000; val_acc: 0.385000
(Iteration 201 / 815) loss: 1.913600
(Iteration 301 / 815) loss: 1.805069
(Epoch 2 / 5) train acc: 0.446000; val_acc: 0.446000
(Iteration 401 / 815) loss: 1.721668
(Epoch 3 / 5) train acc: 0.456000; val_acc: 0.470000
(Iteration 501 / 815) loss: 1.745535
(Iteration 601 / 815) loss: 1.681738
(Epoch 4 / 5) train acc: 0.463000; val_acc: 0.467000
(Iteration 701 / 815) loss: 1.703399
(Iteration 801 / 815) loss: 1.537897
(Epoch 5 / 5) train acc: 0.461000; val_acc: 0.481000
(Iteration 1 / 2450) loss: 2.488635
(Epoch 0 / 5) train acc: 0.123000; val_acc: 0.108000
(Iteration 101 / 2450) loss: 2.072948
(Iteration 201 / 2450) loss: 1.823621
(Iteration 301 / 2450) loss: 1.754690
(Iteration 401 / 2450) loss: 1.864945
(Epoch 1 / 5) train acc: 0.462000; val_acc: 0.447000
(Iteration 501 / 2450) loss: 1.807109
(Iteration 601 / 2450) loss: 1.919490
(Iteration 701 / 2450) loss: 1.751993
(Iteration 801 / 2450) loss: 1.705312
(Iteration 901 / 2450) loss: 1.652280
(Epoch 2 / 5) train acc: 0.490000; val_acc: 0.470000
(Iteration 1001 / 2450) loss: 1.654541
(Iteration 1101 / 2450) loss: 1.637508
(Iteration 1201 / 2450) loss: 1.601131
(Iteration 1301 / 2450) loss: 1.441483
(Iteration 1401 / 2450) loss: 1.459661
(Epoch 3 / 5) train acc: 0.517000; val_acc: 0.485000
(Iteration 1501 / 2450) loss: 1.577307
(Iteration 1601 / 2450) loss: 1.651125
```

```
(Iteration 1701 / 2450) loss: 1.491529
(Iteration 1801 / 2450) loss: 1.399125
(Iteration 1901 / 2450) loss: 1.515892
(Epoch 4 / 5) train acc: 0.534000; val_acc: 0.494000
(Iteration 2001 / 2450) loss: 1.583435
(Iteration 2101 / 2450) loss: 1.507173
(Iteration 2201 / 2450) loss: 1.311818
(Iteration 2301 / 2450) loss: 1.503726
(Iteration 2401 / 2450) loss: 1.611342
(Epoch 5 / 5) train acc: 0.577000; val_acc: 0.511000
(Iteration 1 / 1225) loss: 2.491621
(Epoch 0 / 5) train acc: 0.126000; val_acc: 0.118000
(Iteration 101 / 1225) loss: 2.025437
(Iteration 201 / 1225) loss: 1.862761
(Epoch 1 / 5) train acc: 0.407000; val_acc: 0.402000
(Iteration 301 / 1225) loss: 1.865186
(Iteration 401 / 1225) loss: 1.872641
(Epoch 2 / 5) train acc: 0.435000; val_acc: 0.438000
(Iteration 501 / 1225) loss: 1.783933
(Iteration 601 / 1225) loss: 1.591940
(Iteration 701 / 1225) loss: 1.725353
(Epoch 3 / 5) train acc: 0.471000; val_acc: 0.465000
(Iteration 801 / 1225) loss: 1.689488
(Iteration 901 / 1225) loss: 1.623283
(Epoch 4 / 5) train acc: 0.466000; val_acc: 0.491000
(Iteration 1001 / 1225) loss: 1.598035
(Iteration 1101 / 1225) loss: 1.522084
(Iteration 1201 / 1225) loss: 1.581046
(Epoch 5 / 5) train acc: 0.513000; val_acc: 0.492000
(Iteration 1 / 815) loss: 2.487436
(Epoch 0 / 5) train acc: 0.116000; val_acc: 0.110000
(Iteration 101 / 815) loss: 2.019785
(Epoch 1 / 5) train acc: 0.383000; val_acc: 0.388000
(Iteration 201 / 815) loss: 1.913917
(Iteration 301 / 815) loss: 1.833494
(Epoch 2 / 5) train acc: 0.405000; val_acc: 0.434000
(Iteration 401 / 815) loss: 1.812433
(Epoch 3 / 5) train acc: 0.463000; val_acc: 0.453000
(Iteration 501 / 815) loss: 1.737474
(Iteration 601 / 815) loss: 1.678971
(Epoch 4 / 5) train acc: 0.445000; val_acc: 0.472000
(Iteration 701 / 815) loss: 1.731283
(Iteration 801 / 815) loss: 1.621588
(Epoch 5 / 5) train acc: 0.466000; val_acc: 0.462000
(Iteration 1 / 2450) loss: 2.536335
(Epoch 0 / 5) train acc: 0.131000; val_acc: 0.137000
(Iteration 101 / 2450) loss: 2.208790
(Iteration 201 / 2450) loss: 2.016122
```

```
(Iteration 301 / 2450) loss: 1.807310
(Iteration 401 / 2450) loss: 1.816014
(Epoch 1 / 5) train acc: 0.428000; val_acc: 0.432000
(Iteration 501 / 2450) loss: 1.869305
(Iteration 601 / 2450) loss: 1.649340
(Iteration 701 / 2450) loss: 1.660641
(Iteration 801 / 2450) loss: 1.494285
(Iteration 901 / 2450) loss: 1.601842
(Epoch 2 / 5) train acc: 0.477000; val_acc: 0.492000
(Iteration 1001 / 2450) loss: 1.685475
(Iteration 1101 / 2450) loss: 1.609334
(Iteration 1201 / 2450) loss: 1.670889
(Iteration 1301 / 2450) loss: 1.658920
(Iteration 1401 / 2450) loss: 1.505869
(Epoch 3 / 5) train acc: 0.503000; val_acc: 0.486000
(Iteration 1501 / 2450) loss: 1.615708
(Iteration 1601 / 2450) loss: 1.629695
(Iteration 1701 / 2450) loss: 1.607131
(Iteration 1801 / 2450) loss: 1.523989
(Iteration 1901 / 2450) loss: 1.367948
(Epoch 4 / 5) train acc: 0.522000; val_acc: 0.493000
(Iteration 2001 / 2450) loss: 1.511519
(Iteration 2101 / 2450) loss: 1.576937
(Iteration 2201 / 2450) loss: 1.389908
(Iteration 2301 / 2450) loss: 1.676356
(Iteration 2401 / 2450) loss: 1.291299
(Epoch 5 / 5) train acc: 0.493000; val_acc: 0.516000
(Iteration 1 / 1225) loss: 2.537222
(Epoch 0 / 5) train acc: 0.137000; val_acc: 0.110000
(Iteration 101 / 1225) loss: 2.114229
(Iteration 201 / 1225) loss: 1.951409
(Epoch 1 / 5) train acc: 0.414000; val_acc: 0.400000
(Iteration 301 / 1225) loss: 1.850018
(Iteration 401 / 1225) loss: 1.815727
(Epoch 2 / 5) train acc: 0.455000; val_acc: 0.464000
(Iteration 501 / 1225) loss: 1.770237
(Iteration 601 / 1225) loss: 1.778861
(Iteration 701 / 1225) loss: 1.757818
(Epoch 3 / 5) train acc: 0.508000; val_acc: 0.470000
(Iteration 801 / 1225) loss: 1.757028
(Iteration 901 / 1225) loss: 1.519051
(Epoch 4 / 5) train acc: 0.487000; val_acc: 0.478000
(Iteration 1001 / 1225) loss: 1.741251
(Iteration 1101 / 1225) loss: 1.739874
(Iteration 1201 / 1225) loss: 1.511583
(Epoch 5 / 5) train acc: 0.511000; val_acc: 0.486000
(Iteration 1 / 815) loss: 2.532845
(Epoch 0 / 5) train acc: 0.149000; val_acc: 0.128000
```

```
(Iteration 101 / 815) loss: 2.134559
(Epoch 1 / 5) train acc: 0.385000; val_acc: 0.377000
(Iteration 201 / 815) loss: 1.975948
(Iteration 301 / 815) loss: 1.787858
(Epoch 2 / 5) train acc: 0.412000; val_acc: 0.439000
(Iteration 401 / 815) loss: 1.752773
(Epoch 3 / 5) train acc: 0.466000; val_acc: 0.460000
(Iteration 501 / 815) loss: 1.823636
(Iteration 601 / 815) loss: 1.823791
(Epoch 4 / 5) train acc: 0.467000; val_acc: 0.456000
(Iteration 701 / 815) loss: 1.664921
(Iteration 801 / 815) loss: 1.736316
(Epoch 5 / 5) train acc: 0.492000; val_acc: 0.461000
(Iteration 1 / 2450) loss: 2.578032
(Epoch 0 / 5) train acc: 0.121000; val_acc: 0.133000
(Iteration 101 / 2450) loss: 2.214924
(Iteration 201 / 2450) loss: 1.990281
(Iteration 301 / 2450) loss: 1.961308
(Iteration 401 / 2450) loss: 1.911984
(Epoch 1 / 5) train acc: 0.455000; val_acc: 0.452000
(Iteration 501 / 2450) loss: 1.730637
(Iteration 601 / 2450) loss: 1.709733
(Iteration 701 / 2450) loss: 1.902568
(Iteration 801 / 2450) loss: 1.647081
(Iteration 901 / 2450) loss: 1.724024
(Epoch 2 / 5) train acc: 0.475000; val_acc: 0.459000
(Iteration 1001 / 2450) loss: 1.748678
(Iteration 1101 / 2450) loss: 1.537907
(Iteration 1201 / 2450) loss: 1.550212
(Iteration 1301 / 2450) loss: 1.701807
(Iteration 1401 / 2450) loss: 1.545175
(Epoch 3 / 5) train acc: 0.515000; val_acc: 0.491000
(Iteration 1501 / 2450) loss: 1.521442
(Iteration 1601 / 2450) loss: 1.422490
(Iteration 1701 / 2450) loss: 1.741376
(Iteration 1801 / 2450) loss: 1.527539
(Iteration 1901 / 2450) loss: 1.708815
(Epoch 4 / 5) train acc: 0.526000; val_acc: 0.477000
(Iteration 2001 / 2450) loss: 1.425502
(Iteration 2101 / 2450) loss: 1.375485
(Iteration 2201 / 2450) loss: 1.512257
(Iteration 2301 / 2450) loss: 1.484307
(Iteration 2401 / 2450) loss: 1.357981
(Epoch 5 / 5) train acc: 0.521000; val_acc: 0.525000
(Iteration 1 / 1225) loss: 2.582449
(Epoch 0 / 5) train acc: 0.122000; val_acc: 0.117000
(Iteration 101 / 1225) loss: 2.137701
(Iteration 201 / 1225) loss: 2.012929
```

```
(Epoch 1 / 5) train acc: 0.413000; val_acc: 0.419000
(Iteration 301 / 1225) loss: 1.977018
(Iteration 401 / 1225) loss: 1.883425
(Epoch 2 / 5) train acc: 0.435000; val_acc: 0.448000
(Iteration 501 / 1225) loss: 1.921920
(Iteration 601 / 1225) loss: 1.787620
(Iteration 701 / 1225) loss: 1.669378
(Epoch 3 / 5) train acc: 0.498000; val_acc: 0.461000
(Iteration 801 / 1225) loss: 1.789719
(Iteration 901 / 1225) loss: 1.726677
(Epoch 4 / 5) train acc: 0.470000; val_acc: 0.479000
(Iteration 1001 / 1225) loss: 1.678779
(Iteration 1101 / 1225) loss: 1.647904
(Iteration 1201 / 1225) loss: 1.628802
(Epoch 5 / 5) train acc: 0.498000; val_acc: 0.486000
(Iteration 1 / 815) loss: 2.577033
(Epoch 0 / 5) train acc: 0.115000; val_acc: 0.128000
(Iteration 101 / 815) loss: 2.103048
(Epoch 1 / 5) train acc: 0.407000; val_acc: 0.374000
(Iteration 201 / 815) loss: 2.031363
(Iteration 301 / 815) loss: 1.953919
(Epoch 2 / 5) train acc: 0.467000; val_acc: 0.430000
(Iteration 401 / 815) loss: 1.808954
(Epoch 3 / 5) train acc: 0.444000; val_acc: 0.450000
(Iteration 501 / 815) loss: 1.767319
(Iteration 601 / 815) loss: 1.716730
(Epoch 4 / 5) train acc: 0.470000; val_acc: 0.472000
(Iteration 701 / 815) loss: 1.721299
(Iteration 801 / 815) loss: 1.628019
(Epoch 5 / 5) train acc: 0.510000; val_acc: 0.464000
(Iteration 1 / 2940) loss: 2.439133
(Epoch 0 / 6) train acc: 0.122000; val_acc: 0.117000
(Iteration 101 / 2940) loss: 2.027513
(Iteration 201 / 2940) loss: 1.958378
(Iteration 301 / 2940) loss: 1.811148
(Iteration 401 / 2940) loss: 1.776144
(Epoch 1 / 6) train acc: 0.438000; val_acc: 0.445000
(Iteration 501 / 2940) loss: 1.751419
(Iteration 601 / 2940) loss: 1.742106
(Iteration 701 / 2940) loss: 1.722783
(Iteration 801 / 2940) loss: 1.576535
(Iteration 901 / 2940) loss: 1.585249
(Epoch 2 / 6) train acc: 0.489000; val_acc: 0.460000
(Iteration 1001 / 2940) loss: 1.808273
(Iteration 1101 / 2940) loss: 1.716405
(Iteration 1201 / 2940) loss: 1.737650
(Iteration 1301 / 2940) loss: 1.598207
(Iteration 1401 / 2940) loss: 1.543599
```

```
(Epoch 3 / 6) train acc: 0.527000; val_acc: 0.498000
(Iteration 1501 / 2940) loss: 1.312828
(Iteration 1601 / 2940) loss: 1.550936
(Iteration 1701 / 2940) loss: 1.674899
(Iteration 1801 / 2940) loss: 1.588627
(Iteration 1901 / 2940) loss: 1.448583
(Epoch 4 / 6) train acc: 0.558000; val_acc: 0.506000
(Iteration 2001 / 2940) loss: 1.485132
(Iteration 2101 / 2940) loss: 1.451530
(Iteration 2201 / 2940) loss: 1.364152
(Iteration 2301 / 2940) loss: 1.577086
(Iteration 2401 / 2940) loss: 1.421545
(Epoch 5 / 6) train acc: 0.551000; val_acc: 0.492000
(Iteration 2501 / 2940) loss: 1.536574
(Iteration 2601 / 2940) loss: 1.233748
(Iteration 2701 / 2940) loss: 1.478905
(Iteration 2801 / 2940) loss: 1.410646
(Iteration 2901 / 2940) loss: 1.299132
(Epoch 6 / 6) train acc: 0.578000; val_acc: 0.499000
(Iteration 1 / 1470) loss: 2.440749
(Epoch 0 / 6) train acc: 0.124000; val_acc: 0.133000
(Iteration 101 / 1470) loss: 2.017553
(Iteration 201 / 1470) loss: 2.058500
(Epoch 1 / 6) train acc: 0.416000; val_acc: 0.394000
(Iteration 301 / 1470) loss: 1.779038
(Iteration 401 / 1470) loss: 1.830964
(Epoch 2 / 6) train acc: 0.428000; val_acc: 0.460000
(Iteration 501 / 1470) loss: 1.773009
(Iteration 601 / 1470) loss: 1.622829
(Iteration 701 / 1470) loss: 1.543838
(Epoch 3 / 6) train acc: 0.482000; val_acc: 0.470000
(Iteration 801 / 1470) loss: 1.503258
(Iteration 901 / 1470) loss: 1.578500
(Epoch 4 / 6) train acc: 0.528000; val_acc: 0.479000
(Iteration 1001 / 1470) loss: 1.611297
(Iteration 1101 / 1470) loss: 1.417705
(Iteration 1201 / 1470) loss: 1.408804
(Epoch 5 / 6) train acc: 0.492000; val_acc: 0.483000
(Iteration 1301 / 1470) loss: 1.483279
(Iteration 1401 / 1470) loss: 1.487723
(Epoch 6 / 6) train acc: 0.542000; val_acc: 0.489000
(Iteration 1 / 978) loss: 2.444110
(Epoch 0 / 6) train acc: 0.139000; val_acc: 0.139000
(Iteration 101 / 978) loss: 1.954431
(Epoch 1 / 6) train acc: 0.383000; val_acc: 0.378000
(Iteration 201 / 978) loss: 1.847420
(Iteration 301 / 978) loss: 1.871813
(Epoch 2 / 6) train acc: 0.401000; val_acc: 0.432000
```

```
(Iteration 401 / 978) loss: 1.826086
(Epoch 3 / 6) train acc: 0.467000; val_acc: 0.463000
(Iteration 501 / 978) loss: 1.720403
(Iteration 601 / 978) loss: 1.590835
(Epoch 4 / 6) train acc: 0.465000; val_acc: 0.465000
(Iteration 701 / 978) loss: 1.605801
(Iteration 801 / 978) loss: 1.666878
(Epoch 5 / 6) train acc: 0.491000; val_acc: 0.472000
(Iteration 901 / 978) loss: 1.583947
(Epoch 6 / 6) train acc: 0.512000; val_acc: 0.481000
(Iteration 1 / 2940) loss: 2.485909
(Epoch 0 / 6) train acc: 0.140000; val_acc: 0.160000
(Iteration 101 / 2940) loss: 2.157593
(Iteration 201 / 2940) loss: 1.937737
(Iteration 301 / 2940) loss: 1.720164
(Iteration 401 / 2940) loss: 1.879860
(Epoch 1 / 6) train acc: 0.455000; val_acc: 0.446000
(Iteration 501 / 2940) loss: 1.778545
(Iteration 601 / 2940) loss: 1.750102
(Iteration 701 / 2940) loss: 1.791607
(Iteration 801 / 2940) loss: 1.821049
(Iteration 901 / 2940) loss: 1.481741
(Epoch 2 / 6) train acc: 0.501000; val_acc: 0.483000
(Iteration 1001 / 2940) loss: 1.440872
(Iteration 1101 / 2940) loss: 1.710657
(Iteration 1201 / 2940) loss: 1.546973
(Iteration 1301 / 2940) loss: 1.548134
(Iteration 1401 / 2940) loss: 1.708390
(Epoch 3 / 6) train acc: 0.539000; val_acc: 0.504000
(Iteration 1501 / 2940) loss: 1.719778
(Iteration 1601 / 2940) loss: 1.439302
(Iteration 1701 / 2940) loss: 1.562212
(Iteration 1801 / 2940) loss: 1.440110
(Iteration 1901 / 2940) loss: 1.578607
(Epoch 4 / 6) train acc: 0.518000; val_acc: 0.495000
(Iteration 2001 / 2940) loss: 1.414505
(Iteration 2101 / 2940) loss: 1.435671
(Iteration 2201 / 2940) loss: 1.532156
(Iteration 2301 / 2940) loss: 1.526596
(Iteration 2401 / 2940) loss: 1.455184
(Epoch 5 / 6) train acc: 0.537000; val_acc: 0.515000
(Iteration 2501 / 2940) loss: 1.419176
(Iteration 2601 / 2940) loss: 1.422188
(Iteration 2701 / 2940) loss: 1.350254
(Iteration 2801 / 2940) loss: 1.241638
(Iteration 2901 / 2940) loss: 1.578485
(Epoch 6 / 6) train acc: 0.585000; val_acc: 0.500000
(Iteration 1 / 1470) loss: 2.490597
```

```
(Epoch 0 / 6) train acc: 0.124000; val_acc: 0.107000
(Iteration 101 / 1470) loss: 2.056391
(Iteration 201 / 1470) loss: 1.915334
(Epoch 1 / 6) train acc: 0.399000; val_acc: 0.407000
(Iteration 301 / 1470) loss: 1.789670
(Iteration 401 / 1470) loss: 1.822663
(Epoch 2 / 6) train acc: 0.436000; val_acc: 0.454000
(Iteration 501 / 1470) loss: 1.669402
(Iteration 601 / 1470) loss: 1.555225
(Iteration 701 / 1470) loss: 1.588443
(Epoch 3 / 6) train acc: 0.449000; val_acc: 0.466000
(Iteration 801 / 1470) loss: 1.656628
(Iteration 901 / 1470) loss: 1.555758
(Epoch 4 / 6) train acc: 0.491000; val_acc: 0.474000
(Iteration 1001 / 1470) loss: 1.631317
(Iteration 1101 / 1470) loss: 1.654354
(Iteration 1201 / 1470) loss: 1.456541
(Epoch 5 / 6) train acc: 0.536000; val_acc: 0.482000
(Iteration 1301 / 1470) loss: 1.508504
(Iteration 1401 / 1470) loss: 1.461086
(Epoch 6 / 6) train acc: 0.517000; val_acc: 0.490000
(Iteration 1 / 978) loss: 2.489733
(Epoch 0 / 6) train acc: 0.110000; val_acc: 0.120000
(Iteration 101 / 978) loss: 2.017322
(Epoch 1 / 6) train acc: 0.366000; val_acc: 0.372000
(Iteration 201 / 978) loss: 1.815753
(Iteration 301 / 978) loss: 1.837414
(Epoch 2 / 6) train acc: 0.432000; val_acc: 0.435000
(Iteration 401 / 978) loss: 1.727787
(Epoch 3 / 6) train acc: 0.446000; val_acc: 0.462000
(Iteration 501 / 978) loss: 1.764292
(Iteration 601 / 978) loss: 1.757203
(Epoch 4 / 6) train acc: 0.505000; val_acc: 0.460000
(Iteration 701 / 978) loss: 1.670938
(Iteration 801 / 978) loss: 1.730778
(Epoch 5 / 6) train acc: 0.480000; val_acc: 0.480000
(Iteration 901 / 978) loss: 1.568508
(Epoch 6 / 6) train acc: 0.516000; val_acc: 0.484000
(Iteration 1 / 2940) loss: 2.540082
(Epoch 0 / 6) train acc: 0.116000; val_acc: 0.100000
(Iteration 101 / 2940) loss: 2.013344
(Iteration 201 / 2940) loss: 2.015147
(Iteration 301 / 2940) loss: 2.016735
(Iteration 401 / 2940) loss: 1.865175
(Epoch 1 / 6) train acc: 0.439000; val_acc: 0.445000
(Iteration 501 / 2940) loss: 1.672199
(Iteration 601 / 2940) loss: 1.683219
(Iteration 701 / 2940) loss: 1.723121
```

```
(Iteration 801 / 2940) loss: 1.512800
(Iteration 901 / 2940) loss: 1.610056
(Epoch 2 / 6) train acc: 0.497000; val_acc: 0.465000
(Iteration 1001 / 2940) loss: 1.698668
(Iteration 1101 / 2940) loss: 1.539639
(Iteration 1201 / 2940) loss: 1.515639
(Iteration 1301 / 2940) loss: 1.751209
(Iteration 1401 / 2940) loss: 1.490312
(Epoch 3 / 6) train acc: 0.540000; val_acc: 0.483000
(Iteration 1501 / 2940) loss: 1.389063
(Iteration 1601 / 2940) loss: 1.607137
(Iteration 1701 / 2940) loss: 1.576806
(Iteration 1801 / 2940) loss: 1.576790
(Iteration 1901 / 2940) loss: 1.624738
(Epoch 4 / 6) train acc: 0.538000; val_acc: 0.525000
(Iteration 2001 / 2940) loss: 1.571833
(Iteration 2101 / 2940) loss: 1.705258
(Iteration 2201 / 2940) loss: 1.650332
(Iteration 2301 / 2940) loss: 1.607795
(Iteration 2401 / 2940) loss: 1.639728
(Epoch 5 / 6) train acc: 0.558000; val_acc: 0.505000
(Iteration 2501 / 2940) loss: 1.576483
(Iteration 2601 / 2940) loss: 1.441636
(Iteration 2701 / 2940) loss: 1.430283
(Iteration 2801 / 2940) loss: 1.448516
(Iteration 2901 / 2940) loss: 1.401534
(Epoch 6 / 6) train acc: 0.566000; val_acc: 0.504000
(Iteration 1 / 1470) loss: 2.533657
(Epoch 0 / 6) train acc: 0.116000; val_acc: 0.134000
(Iteration 101 / 1470) loss: 2.109186
(Iteration 201 / 1470) loss: 1.968264
(Epoch 1 / 6) train acc: 0.389000; val_acc: 0.415000
(Iteration 301 / 1470) loss: 1.831445
(Iteration 401 / 1470) loss: 1.868483
(Epoch 2 / 6) train acc: 0.458000; val_acc: 0.462000
(Iteration 501 / 1470) loss: 1.762201
(Iteration 601 / 1470) loss: 1.652081
(Iteration 701 / 1470) loss: 1.724642
(Epoch 3 / 6) train acc: 0.484000; val_acc: 0.474000
(Iteration 801 / 1470) loss: 1.638343
(Iteration 901 / 1470) loss: 1.712372
(Epoch 4 / 6) train acc: 0.498000; val_acc: 0.491000
(Iteration 1001 / 1470) loss: 1.734612
(Iteration 1101 / 1470) loss: 1.711522
(Iteration 1201 / 1470) loss: 1.519453
(Epoch 5 / 6) train acc: 0.524000; val_acc: 0.484000
(Iteration 1301 / 1470) loss: 1.520908
(Iteration 1401 / 1470) loss: 1.500242
```

```
(Epoch 6 / 6) train acc: 0.527000; val_acc: 0.490000
(Iteration 1 / 978) loss: 2.538003
(Epoch 0 / 6) train acc: 0.143000; val_acc: 0.148000
(Iteration 101 / 978) loss: 2.050454
(Epoch 1 / 6) train acc: 0.367000; val_acc: 0.360000
(Iteration 201 / 978) loss: 1.995735
(Iteration 301 / 978) loss: 1.878957
(Epoch 2 / 6) train acc: 0.421000; val_acc: 0.436000
(Iteration 401 / 978) loss: 1.863864
(Epoch 3 / 6) train acc: 0.456000; val_acc: 0.457000
(Iteration 501 / 978) loss: 1.850376
(Iteration 601 / 978) loss: 1.810732
(Epoch 4 / 6) train acc: 0.503000; val_acc: 0.473000
(Iteration 701 / 978) loss: 1.659837
(Iteration 801 / 978) loss: 1.689495
(Epoch 5 / 6) train acc: 0.482000; val_acc: 0.471000
(Iteration 901 / 978) loss: 1.689639
(Epoch 6 / 6) train acc: 0.495000; val_acc: 0.491000
(Iteration 1 / 2940) loss: 2.580179
(Epoch 0 / 6) train acc: 0.144000; val_acc: 0.153000
(Iteration 101 / 2940) loss: 2.159556
(Iteration 201 / 2940) loss: 1.915744
(Iteration 301 / 2940) loss: 1.925321
(Iteration 401 / 2940) loss: 1.825170
(Epoch 1 / 6) train acc: 0.444000; val_acc: 0.435000
(Iteration 501 / 2940) loss: 1.972752
(Iteration 601 / 2940) loss: 1.776105
(Iteration 701 / 2940) loss: 1.571112
(Iteration 801 / 2940) loss: 1.832094
(Iteration 901 / 2940) loss: 1.980723
(Epoch 2 / 6) train acc: 0.495000; val_acc: 0.467000
(Iteration 1001 / 2940) loss: 1.557527
(Iteration 1101 / 2940) loss: 1.692713
(Iteration 1201 / 2940) loss: 1.642812
(Iteration 1301 / 2940) loss: 1.642384
(Iteration 1401 / 2940) loss: 1.635060
(Epoch 3 / 6) train acc: 0.509000; val_acc: 0.482000
(Iteration 1501 / 2940) loss: 1.618557
(Iteration 1601 / 2940) loss: 1.499853
(Iteration 1701 / 2940) loss: 1.719790
(Iteration 1801 / 2940) loss: 1.400576
(Iteration 1901 / 2940) loss: 1.462080
(Epoch 4 / 6) train acc: 0.534000; val_acc: 0.502000
(Iteration 2001 / 2940) loss: 1.701857
(Iteration 2101 / 2940) loss: 1.554307
(Iteration 2201 / 2940) loss: 1.711947
(Iteration 2301 / 2940) loss: 1.618801
(Iteration 2401 / 2940) loss: 1.465403
```

```
(Epoch 5 / 6) train acc: 0.552000; val_acc: 0.508000
(Iteration 2501 / 2940) loss: 1.394680
(Iteration 2601 / 2940) loss: 1.439718
(Iteration 2701 / 2940) loss: 1.423988
(Iteration 2801 / 2940) loss: 1.423051
(Iteration 2901 / 2940) loss: 1.478659
(Epoch 6 / 6) train acc: 0.549000; val_acc: 0.521000
(Iteration 1 / 1470) loss: 2.577577
(Epoch 0 / 6) train acc: 0.129000; val_acc: 0.130000
(Iteration 101 / 1470) loss: 2.252343
(Iteration 201 / 1470) loss: 2.017732
(Epoch 1 / 6) train acc: 0.407000; val_acc: 0.410000
(Iteration 301 / 1470) loss: 1.798777
(Iteration 401 / 1470) loss: 1.961303
(Epoch 2 / 6) train acc: 0.462000; val_acc: 0.448000
(Iteration 501 / 1470) loss: 1.712374
(Iteration 601 / 1470) loss: 1.814703
(Iteration 701 / 1470) loss: 1.763791
(Epoch 3 / 6) train acc: 0.471000; val_acc: 0.465000
(Iteration 801 / 1470) loss: 1.669711
(Iteration 901 / 1470) loss: 1.727432
(Epoch 4 / 6) train acc: 0.481000; val_acc: 0.485000
(Iteration 1001 / 1470) loss: 1.654010
(Iteration 1101 / 1470) loss: 1.644945
(Iteration 1201 / 1470) loss: 1.550190
(Epoch 5 / 6) train acc: 0.537000; val_acc: 0.483000
(Iteration 1301 / 1470) loss: 1.642527
(Iteration 1401 / 1470) loss: 1.613523
(Epoch 6 / 6) train acc: 0.511000; val_acc: 0.489000
(Iteration 1 / 978) loss: 2.580262
(Epoch 0 / 6) train acc: 0.116000; val_acc: 0.141000
(Iteration 101 / 978) loss: 2.124639
(Epoch 1 / 6) train acc: 0.376000; val_acc: 0.369000
(Iteration 201 / 978) loss: 1.963701
(Iteration 301 / 978) loss: 1.990271
(Epoch 2 / 6) train acc: 0.410000; val_acc: 0.425000
(Iteration 401 / 978) loss: 1.778087
(Epoch 3 / 6) train acc: 0.446000; val_acc: 0.466000
(Iteration 501 / 978) loss: 1.736038
(Iteration 601 / 978) loss: 1.742618
(Epoch 4 / 6) train acc: 0.465000; val_acc: 0.462000
(Iteration 701 / 978) loss: 1.661690
(Iteration 801 / 978) loss: 1.590750
(Epoch 5 / 6) train acc: 0.467000; val_acc: 0.462000
(Iteration 901 / 978) loss: 1.645271
(Epoch 6 / 6) train acc: 0.476000; val_acc: 0.474000
(Iteration 1 / 1960) loss: 2.346627
(Epoch 0 / 4) train acc: 0.154000; val_acc: 0.131000
```

```
/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/layers.py:721:
RuntimeWarning: divide by zero encountered in log
  loss = (-np.sum(np.log(probs[np.arange(x.shape[0]), y])))/num_train

(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.122000; val_acc: 0.123000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf

/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/layers.py:25:
RuntimeWarning: overflow encountered in matmul
  out = x.reshape(x.shape[0], -1) @ w + b
/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/layers.py:719:
RuntimeWarning: overflow encountered in subtract
  scores_prob = x - np.max(x, axis=1, keepdims=True)
/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/layers.py:719:
RuntimeWarning: invalid value encountered in subtract
  scores_prob = x - np.max(x, axis=1, keepdims=True)

(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.096000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.104000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
(Epoch 4 / 4) train acc: 0.098000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.351633
(Epoch 0 / 4) train acc: 0.187000; val_acc: 0.204000
(Iteration 101 / 980) loss: inf
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.107000; val_acc: 0.092000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
(Epoch 2 / 4) train acc: 0.097000; val_acc: 0.090000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
```

```
(Epoch 3 / 4) train acc: 0.087000; val_acc: 0.102000
(Iteration 801 / 980) loss: inf
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.089000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.352004
(Epoch 0 / 4) train acc: 0.185000; val_acc: 0.186000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.081000; val_acc: 0.059000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.073000; val_acc: 0.059000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.117000; val_acc: 0.118000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.065000; val_acc: 0.057000
(Iteration 1 / 1960) loss: 2.371855
(Epoch 0 / 4) train acc: 0.170000; val_acc: 0.173000
(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.074000; val_acc: 0.059000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf
(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.116000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.086000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
(Epoch 4 / 4) train acc: 0.090000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.361205
(Epoch 0 / 4) train acc: 0.183000; val_acc: 0.207000
(Iteration 101 / 980) loss: inf
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.079000; val_acc: 0.085000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
```

```
(Epoch 2 / 4) train acc: 0.104000; val_acc: 0.102000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
(Epoch 3 / 4) train acc: 0.110000; val_acc: 0.105000
(Iteration 801 / 980) loss: inf
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.088000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.363648
(Epoch 0 / 4) train acc: 0.152000; val_acc: 0.170000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.101000; val_acc: 0.115000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.124000; val_acc: 0.135000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.098000; val_acc: 0.094000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.077000; val_acc: 0.061000
(Iteration 1 / 1960) loss: 2.382859
(Epoch 0 / 4) train acc: 0.180000; val_acc: 0.194000
(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.127000; val_acc: 0.113000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf
(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.090000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.093000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
(Epoch 4 / 4) train acc: 0.098000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.376710
(Epoch 0 / 4) train acc: 0.156000; val_acc: 0.166000
(Iteration 101 / 980) loss: inf
```

```
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.116000; val_acc: 0.094000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
(Epoch 2 / 4) train acc: 0.117000; val_acc: 0.104000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
(Epoch 3 / 4) train acc: 0.074000; val_acc: 0.085000
(Iteration 801 / 980) loss: inf
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.099000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.381208
(Epoch 0 / 4) train acc: 0.211000; val_acc: 0.194000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.098000; val_acc: 0.106000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.090000; val_acc: 0.082000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.124000; val_acc: 0.122000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.130000; val_acc: 0.131000
(Iteration 1 / 1960) loss: 2.394566
(Epoch 0 / 4) train acc: 0.136000; val_acc: 0.144000
(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.043000; val_acc: 0.044000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf
(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.103000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.093000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
```

```
(Epoch 4 / 4) train acc: 0.097000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.395055
(Epoch 0 / 4) train acc: 0.135000; val_acc: 0.150000
(Iteration 101 / 980) loss: inf
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.095000; val_acc: 0.105000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
(Epoch 2 / 4) train acc: 0.130000; val_acc: 0.132000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
(Epoch 3 / 4) train acc: 0.100000; val_acc: 0.099000
(Iteration 801 / 980) loss: inf
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.104000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.392751
(Epoch 0 / 4) train acc: 0.143000; val_acc: 0.167000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.103000; val_acc: 0.122000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.096000; val_acc: 0.069000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.107000; val_acc: 0.120000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.057000; val_acc: 0.073000
(Iteration 1 / 2450) loss: 2.344866
(Epoch 0 / 5) train acc: 0.182000; val_acc: 0.187000
(Iteration 101 / 2450) loss: inf
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.076000; val_acc: 0.082000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
(Epoch 2 / 5) train acc: 0.097000; val_acc: 0.087000
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
(Iteration 1301 / 2450) loss: nan
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.110000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
```

```
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.104000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
(Epoch 5 / 5) train acc: 0.100000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.348914
(Epoch 0 / 5) train acc: 0.190000; val_acc: 0.205000
(Iteration 101 / 1225) loss: inf
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.114000; val_acc: 0.121000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.088000; val_acc: 0.104000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.118000; val_acc: 0.113000
(Iteration 801 / 1225) loss: inf
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.095000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.107000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.354058
(Epoch 0 / 5) train acc: 0.191000; val_acc: 0.199000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.089000; val_acc: 0.081000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.092000; val_acc: 0.090000
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.112000; val_acc: 0.110000
(Iteration 501 / 815) loss: inf
(Iteration 601 / 815) loss: inf
(Epoch 4 / 5) train acc: 0.084000; val_acc: 0.096000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
(Epoch 5 / 5) train acc: 0.094000; val_acc: 0.071000
(Iteration 1 / 2450) loss: 2.363431
(Epoch 0 / 5) train acc: 0.196000; val_acc: 0.193000
(Iteration 101 / 2450) loss: inf
```

```
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.115000; val_acc: 0.120000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
(Epoch 2 / 5) train acc: 0.110000; val_acc: 0.087000
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
(Iteration 1301 / 2450) loss: nan
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.099000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.090000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
(Epoch 5 / 5) train acc: 0.109000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.364457
(Epoch 0 / 5) train acc: 0.148000; val_acc: 0.175000
(Iteration 101 / 1225) loss: inf
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.106000; val_acc: 0.106000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.104000; val_acc: 0.133000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.097000; val_acc: 0.097000
(Iteration 801 / 1225) loss: inf
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.100000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.101000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.367541
```

```
(Epoch 0 / 5) train acc: 0.178000; val_acc: 0.163000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.100000; val_acc: 0.085000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.144000; val_acc: 0.109000
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.112000; val_acc: 0.102000
(Iteration 501 / 815) loss: inf
(Iteration 601 / 815) loss: inf
(Epoch 4 / 5) train acc: 0.092000; val_acc: 0.054000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
(Epoch 5 / 5) train acc: 0.139000; val_acc: 0.138000
(Iteration 1 / 2450) loss: 2.383644
(Epoch 0 / 5) train acc: 0.140000; val_acc: 0.141000
(Iteration 101 / 2450) loss: inf
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.063000; val_acc: 0.044000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
(Epoch 2 / 5) train acc: 0.122000; val_acc: 0.087000
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
(Iteration 1301 / 2450) loss: nan
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.116000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.088000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
(Epoch 5 / 5) train acc: 0.114000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.378171
(Epoch 0 / 5) train acc: 0.174000; val_acc: 0.175000
(Iteration 101 / 1225) loss: inf
```

```
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.072000; val_acc: 0.061000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.079000; val_acc: 0.057000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.065000; val_acc: 0.057000
(Iteration 801 / 1225) loss: inf
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.096000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.090000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.380288
(Epoch 0 / 5) train acc: 0.155000; val_acc: 0.181000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.094000; val_acc: 0.102000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.086000; val_acc: 0.087000
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.122000; val_acc: 0.142000
(Iteration 501 / 815) loss: inf
(Iteration 601 / 815) loss: inf
(Epoch 4 / 5) train acc: 0.116000; val_acc: 0.116000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
(Epoch 5 / 5) train acc: 0.073000; val_acc: 0.097000
(Iteration 1 / 2450) loss: 2.395361
(Epoch 0 / 5) train acc: 0.173000; val_acc: 0.169000
(Iteration 101 / 2450) loss: inf
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.103000; val_acc: 0.119000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
(Epoch 2 / 5) train acc: 0.104000; val_acc: 0.087000
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
(Iteration 1301 / 2450) loss: nan
```

```
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.080000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.088000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
(Epoch 5 / 5) train acc: 0.106000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.395227
(Epoch 0 / 5) train acc: 0.124000; val_acc: 0.148000
(Iteration 101 / 1225) loss: inf
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.106000; val_acc: 0.094000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.059000; val_acc: 0.065000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.113000; val_acc: 0.126000
(Iteration 801 / 1225) loss: inf
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.114000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.099000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.396664
(Epoch 0 / 5) train acc: 0.221000; val_acc: 0.230000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.101000; val_acc: 0.099000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.061000; val_acc: 0.051000
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.053000; val_acc: 0.059000
(Iteration 501 / 815) loss: inf
(Iteration 601 / 815) loss: inf
(Epoch 4 / 5) train acc: 0.123000; val_acc: 0.087000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
(Epoch 5 / 5) train acc: 0.078000; val_acc: 0.089000
```

```
(Iteration 1 / 2940) loss: 2.349098
(Epoch 0 / 6) train acc: 0.146000; val_acc: 0.178000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.107000; val_acc: 0.097000
(Iteration 501 / 2940) loss: inf
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
(Epoch 2 / 6) train acc: 0.103000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.092000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.101000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.111000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
(Epoch 6 / 6) train acc: 0.105000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.348627
(Epoch 0 / 6) train acc: 0.189000; val_acc: 0.159000
(Iteration 101 / 1470) loss: inf
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.102000; val_acc: 0.104000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
(Epoch 2 / 6) train acc: 0.098000; val_acc: 0.124000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
```

```
(Epoch 3 / 6) train acc: 0.121000; val_acc: 0.118000
(Iteration 801 / 1470) loss: inf
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.113000; val_acc: 0.087000
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.120000; val_acc: 0.087000
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.113000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.348786
(Epoch 0 / 6) train acc: 0.174000; val_acc: 0.181000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.114000; val_acc: 0.109000
(Iteration 201 / 978) loss: inf
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.100000; val_acc: 0.098000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.109000; val_acc: 0.136000
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.067000; val_acc: 0.089000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.152000; val_acc: 0.135000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.092000; val_acc: 0.087000
(Iteration 1 / 2940) loss: 2.362925
(Epoch 0 / 6) train acc: 0.206000; val_acc: 0.220000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.104000; val_acc: 0.108000
(Iteration 501 / 2940) loss: inf
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
(Epoch 2 / 6) train acc: 0.105000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.103000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
```

```
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.088000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.104000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
(Epoch 6 / 6) train acc: 0.108000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.360583
(Epoch 0 / 6) train acc: 0.185000; val_acc: 0.200000
(Iteration 101 / 1470) loss: inf
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.086000; val_acc: 0.073000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
(Epoch 2 / 6) train acc: 0.107000; val_acc: 0.094000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
(Epoch 3 / 6) train acc: 0.132000; val_acc: 0.138000
(Iteration 801 / 1470) loss: inf
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.118000; val_acc: 0.087000
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.109000; val_acc: 0.087000
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.102000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.365247
(Epoch 0 / 6) train acc: 0.226000; val_acc: 0.227000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.105000; val_acc: 0.113000
(Iteration 201 / 978) loss: inf
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.115000; val_acc: 0.109000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.100000; val_acc: 0.080000
```

```
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.092000; val_acc: 0.084000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.141000; val_acc: 0.129000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.107000; val_acc: 0.100000
(Iteration 1 / 2940) loss: 2.380794
(Epoch 0 / 6) train acc: 0.195000; val_acc: 0.178000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.084000; val_acc: 0.104000
(Iteration 501 / 2940) loss: inf
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
(Epoch 2 / 6) train acc: 0.094000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.111000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.117000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.101000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
(Epoch 6 / 6) train acc: 0.116000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.381037
(Epoch 0 / 6) train acc: 0.169000; val_acc: 0.172000
(Iteration 101 / 1470) loss: inf
```

```
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.095000; val_acc: 0.115000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
(Epoch 2 / 6) train acc: 0.100000; val_acc: 0.091000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
(Epoch 3 / 6) train acc: 0.079000; val_acc: 0.083000
(Iteration 801 / 1470) loss: inf
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.088000; val_acc: 0.087000
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.073000; val_acc: 0.087000
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.102000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.382648
(Epoch 0 / 6) train acc: 0.188000; val_acc: 0.214000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.106000; val_acc: 0.112000
(Iteration 201 / 978) loss: inf
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.119000; val_acc: 0.109000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.141000; val_acc: 0.139000
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.128000; val_acc: 0.106000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.083000; val_acc: 0.076000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.102000; val_acc: 0.087000
(Iteration 1 / 2940) loss: 2.395471
(Epoch 0 / 6) train acc: 0.175000; val_acc: 0.179000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.111000; val_acc: 0.120000
(Iteration 501 / 2940) loss: inf
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
```

```
(Epoch 2 / 6) train acc: 0.110000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.104000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.103000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.092000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
(Epoch 6 / 6) train acc: 0.118000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.394694
(Epoch 0 / 6) train acc: 0.145000; val_acc: 0.156000
(Iteration 101 / 1470) loss: inf
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.103000; val_acc: 0.093000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
(Epoch 2 / 6) train acc: 0.103000; val_acc: 0.079000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
(Epoch 3 / 6) train acc: 0.134000; val_acc: 0.118000
(Iteration 801 / 1470) loss: inf
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.104000; val_acc: 0.087000
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.098000; val_acc: 0.087000
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.090000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.398438
```

```
(Epoch 0 / 6) train acc: 0.172000; val_acc: 0.169000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.120000; val_acc: 0.109000
(Iteration 201 / 978) loss: inf
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.139000; val_acc: 0.126000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.127000; val_acc: 0.124000
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.157000; val_acc: 0.140000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.098000; val_acc: 0.074000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.097000; val_acc: 0.087000
(Iteration 1 / 1960) loss: 2.398789
(Epoch 0 / 4) train acc: 0.196000; val_acc: 0.210000
(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.067000; val_acc: 0.066000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf
(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.109000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.106000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
(Epoch 4 / 4) train acc: 0.105000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.396335
(Epoch 0 / 4) train acc: 0.194000; val_acc: 0.186000
(Iteration 101 / 980) loss: inf
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.082000; val_acc: 0.078000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
```

```
(Epoch 2 / 4) train acc: 0.117000; val_acc: 0.126000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
(Epoch 3 / 4) train acc: 0.094000; val_acc: 0.095000
(Iteration 801 / 980) loss: nan
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.102000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.399573
(Epoch 0 / 4) train acc: 0.165000; val_acc: 0.175000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.129000; val_acc: 0.144000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.124000; val_acc: 0.112000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.059000; val_acc: 0.076000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.106000; val_acc: 0.103000
(Iteration 1 / 1960) loss: 2.427710
(Epoch 0 / 4) train acc: 0.184000; val_acc: 0.178000
(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.077000; val_acc: 0.093000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf
(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.098000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.086000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
(Epoch 4 / 4) train acc: 0.091000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.425995
(Epoch 0 / 4) train acc: 0.185000; val_acc: 0.172000
(Iteration 101 / 980) loss: inf
```

```
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.128000; val_acc: 0.124000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
(Epoch 2 / 4) train acc: 0.125000; val_acc: 0.117000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
(Epoch 3 / 4) train acc: 0.145000; val_acc: 0.149000
(Iteration 801 / 980) loss: inf
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.105000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.423307
(Epoch 0 / 4) train acc: 0.176000; val_acc: 0.186000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.058000; val_acc: 0.047000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.087000; val_acc: 0.095000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.074000; val_acc: 0.072000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.086000; val_acc: 0.092000
(Iteration 1 / 1960) loss: 2.463908
(Epoch 0 / 4) train acc: 0.174000; val_acc: 0.188000
(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.103000; val_acc: 0.070000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf
(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.091000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.096000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
```

```
(Epoch 4 / 4) train acc: 0.115000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.459870
(Epoch 0 / 4) train acc: 0.192000; val_acc: 0.209000
(Iteration 101 / 980) loss: inf
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.151000; val_acc: 0.156000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
(Epoch 2 / 4) train acc: 0.122000; val_acc: 0.090000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
(Epoch 3 / 4) train acc: 0.095000; val_acc: 0.085000
(Iteration 801 / 980) loss: inf
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.095000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.455848
(Epoch 0 / 4) train acc: 0.190000; val_acc: 0.196000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.089000; val_acc: 0.114000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.080000; val_acc: 0.070000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.065000; val_acc: 0.087000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.102000; val_acc: 0.113000
(Iteration 1 / 1960) loss: 2.490396
(Epoch 0 / 4) train acc: 0.150000; val_acc: 0.141000
(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.134000; val_acc: 0.133000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf
(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.089000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.084000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
```

```
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
(Epoch 4 / 4) train acc: 0.108000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.492056
(Epoch 0 / 4) train acc: 0.170000; val_acc: 0.175000
(Iteration 101 / 980) loss: inf
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.064000; val_acc: 0.048000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
(Epoch 2 / 4) train acc: 0.074000; val_acc: 0.086000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
(Epoch 3 / 4) train acc: 0.104000; val_acc: 0.106000
(Iteration 801 / 980) loss: inf
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.098000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.486207
(Epoch 0 / 4) train acc: 0.179000; val_acc: 0.178000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.054000; val_acc: 0.065000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.110000; val_acc: 0.087000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.054000; val_acc: 0.048000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.070000; val_acc: 0.064000
(Iteration 1 / 2450) loss: 2.396360
(Epoch 0 / 5) train acc: 0.171000; val_acc: 0.179000
(Iteration 101 / 2450) loss: inf
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.112000; val_acc: 0.100000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
(Epoch 2 / 5) train acc: 0.097000; val_acc: 0.087000
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
```

```
(Iteration 1301 / 2450) loss: nan
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.104000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.100000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
(Epoch 5 / 5) train acc: 0.102000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.398433
(Epoch 0 / 5) train acc: 0.194000; val_acc: 0.209000
(Iteration 101 / 1225) loss: inf
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.076000; val_acc: 0.075000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.065000; val_acc: 0.062000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.078000; val_acc: 0.066000
(Iteration 801 / 1225) loss: inf
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.095000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.106000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.399269
(Epoch 0 / 5) train acc: 0.203000; val_acc: 0.229000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.090000; val_acc: 0.098000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.087000; val_acc: 0.084000
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.117000; val_acc: 0.107000
(Iteration 501 / 815) loss: inf
(Iteration 601 / 815) loss: inf
(Epoch 4 / 5) train acc: 0.141000; val_acc: 0.153000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
```

```
(Epoch 5 / 5) train acc: 0.090000; val_acc: 0.099000
(Iteration 1 / 2450) loss: 2.432522
(Epoch 0 / 5) train acc: 0.184000; val_acc: 0.197000
(Iteration 101 / 2450) loss: inf
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.086000; val_acc: 0.101000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
(Epoch 2 / 5) train acc: 0.099000; val_acc: 0.087000
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
(Iteration 1301 / 2450) loss: nan
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.099000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.106000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
(Epoch 5 / 5) train acc: 0.090000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.422874
(Epoch 0 / 5) train acc: 0.228000; val_acc: 0.209000
(Iteration 101 / 1225) loss: inf
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.163000; val_acc: 0.134000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.101000; val_acc: 0.087000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.101000; val_acc: 0.099000
(Iteration 801 / 1225) loss: inf
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.097000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
```

```
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.097000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.427749
(Epoch 0 / 5) train acc: 0.189000; val_acc: 0.227000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.095000; val_acc: 0.085000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.101000; val_acc: 0.120000
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.159000; val_acc: 0.130000
(Iteration 501 / 815) loss: inf
(Iteration 601 / 815) loss: inf
(Epoch 4 / 5) train acc: 0.100000; val_acc: 0.086000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
(Epoch 5 / 5) train acc: 0.088000; val_acc: 0.089000
(Iteration 1 / 2450) loss: 2.467783
(Epoch 0 / 5) train acc: 0.128000; val_acc: 0.118000
(Iteration 101 / 2450) loss: inf
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.081000; val_acc: 0.092000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
(Epoch 2 / 5) train acc: 0.101000; val_acc: 0.087000
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
(Iteration 1301 / 2450) loss: nan
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.099000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.107000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
```

```
(Epoch 5 / 5) train acc: 0.091000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.459344
(Epoch 0 / 5) train acc: 0.200000; val_acc: 0.219000
(Iteration 101 / 1225) loss: inf
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.101000; val_acc: 0.086000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.104000; val_acc: 0.107000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.057000; val_acc: 0.050000
(Iteration 801 / 1225) loss: inf
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.093000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.119000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.455592
(Epoch 0 / 5) train acc: 0.157000; val_acc: 0.164000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.090000; val_acc: 0.099000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.077000; val_acc: 0.094000
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.077000; val_acc: 0.083000
(Iteration 501 / 815) loss: inf
(Iteration 601 / 815) loss: inf
(Epoch 4 / 5) train acc: 0.107000; val_acc: 0.096000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
(Epoch 5 / 5) train acc: 0.069000; val_acc: 0.072000
(Iteration 1 / 2450) loss: 2.484966
(Epoch 0 / 5) train acc: 0.172000; val_acc: 0.154000
(Iteration 101 / 2450) loss: inf
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.090000; val_acc: 0.081000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
(Epoch 2 / 5) train acc: 0.102000; val_acc: 0.087000
```

```
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
(Iteration 1301 / 2450) loss: nan
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.097000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.078000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
(Epoch 5 / 5) train acc: 0.096000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.490841
(Epoch 0 / 5) train acc: 0.215000; val_acc: 0.190000
(Iteration 101 / 1225) loss: inf
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.079000; val_acc: 0.096000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.063000; val_acc: 0.064000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.104000; val_acc: 0.095000
(Iteration 801 / 1225) loss: nan
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.097000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.104000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.487876
(Epoch 0 / 5) train acc: 0.221000; val_acc: 0.204000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.117000; val_acc: 0.126000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.108000; val_acc: 0.118000
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.105000; val_acc: 0.098000
(Iteration 501 / 815) loss: inf
(Iteration 601 / 815) loss: inf
```

```
(Epoch 4 / 5) train acc: 0.111000; val_acc: 0.112000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
(Epoch 5 / 5) train acc: 0.116000; val_acc: 0.100000
(Iteration 1 / 2940) loss: 2.389266
(Epoch 0 / 6) train acc: 0.161000; val_acc: 0.182000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.078000; val_acc: 0.085000
(Iteration 501 / 2940) loss: inf
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
(Epoch 2 / 6) train acc: 0.092000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.099000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.098000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.104000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
(Epoch 6 / 6) train acc: 0.091000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.393821
(Epoch 0 / 6) train acc: 0.192000; val_acc: 0.193000
(Iteration 101 / 1470) loss: inf
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.116000; val_acc: 0.110000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
```

```
(Epoch 2 / 6) train acc: 0.078000; val_acc: 0.080000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
(Epoch 3 / 6) train acc: 0.074000; val_acc: 0.053000
(Iteration 801 / 1470) loss: inf
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.106000; val_acc: 0.087000
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.100000; val_acc: 0.087000
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.092000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.396834
(Epoch 0 / 6) train acc: 0.210000; val_acc: 0.194000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.078000; val_acc: 0.099000
(Iteration 201 / 978) loss: inf
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.118000; val_acc: 0.102000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.107000; val_acc: 0.084000
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.086000; val_acc: 0.089000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.139000; val_acc: 0.178000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.137000; val_acc: 0.132000
(Iteration 1 / 2940) loss: 2.425520
(Epoch 0 / 6) train acc: 0.183000; val_acc: 0.199000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.117000; val_acc: 0.094000
(Iteration 501 / 2940) loss: inf
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
(Epoch 2 / 6) train acc: 0.122000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
```

```
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.110000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.106000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.107000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
(Epoch 6 / 6) train acc: 0.088000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.426202
(Epoch 0 / 6) train acc: 0.188000; val_acc: 0.181000
(Iteration 101 / 1470) loss: inf
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.091000; val_acc: 0.075000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
(Epoch 2 / 6) train acc: 0.115000; val_acc: 0.103000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
(Epoch 3 / 6) train acc: 0.098000; val_acc: 0.120000
(Iteration 801 / 1470) loss: inf
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.098000; val_acc: 0.087000
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.104000; val_acc: 0.087000
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.090000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.428732
(Epoch 0 / 6) train acc: 0.175000; val_acc: 0.172000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.084000; val_acc: 0.113000
(Iteration 201 / 978) loss: inf
```

```
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.125000; val_acc: 0.139000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.053000; val_acc: 0.042000
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.113000; val_acc: 0.088000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.076000; val_acc: 0.085000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.104000; val_acc: 0.087000
(Iteration 1 / 2940) loss: 2.463074
(Epoch 0 / 6) train acc: 0.175000; val_acc: 0.169000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.094000; val_acc: 0.085000
(Iteration 501 / 2940) loss: inf
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
(Epoch 2 / 6) train acc: 0.102000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.118000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.114000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.107000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
```

```
(Epoch 6 / 6) train acc: 0.092000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.463639
(Epoch 0 / 6) train acc: 0.211000; val_acc: 0.212000
(Iteration 101 / 1470) loss: inf
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.112000; val_acc: 0.104000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
(Epoch 2 / 6) train acc: 0.081000; val_acc: 0.073000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
(Epoch 3 / 6) train acc: 0.086000; val_acc: 0.100000
(Iteration 801 / 1470) loss: inf
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.104000; val_acc: 0.087000
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.096000; val_acc: 0.087000
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.123000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.456524
(Epoch 0 / 6) train acc: 0.189000; val_acc: 0.197000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.089000; val_acc: 0.117000
(Iteration 201 / 978) loss: inf
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.071000; val_acc: 0.063000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.078000; val_acc: 0.069000
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.101000; val_acc: 0.097000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.086000; val_acc: 0.099000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.102000; val_acc: 0.087000
(Iteration 1 / 2940) loss: 2.482324
(Epoch 0 / 6) train acc: 0.190000; val_acc: 0.167000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.145000; val_acc: 0.149000
(Iteration 501 / 2940) loss: inf
```

```
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
(Epoch 2 / 6) train acc: 0.126000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.101000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.108000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.095000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
(Epoch 6 / 6) train acc: 0.101000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.489144
(Epoch 0 / 6) train acc: 0.205000; val_acc: 0.186000
(Iteration 101 / 1470) loss: inf
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.093000; val_acc: 0.095000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
(Epoch 2 / 6) train acc: 0.080000; val_acc: 0.090000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
(Epoch 3 / 6) train acc: 0.067000; val_acc: 0.085000
(Iteration 801 / 1470) loss: inf
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.095000; val_acc: 0.087000
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.107000; val_acc: 0.087000
```

```
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.098000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.488859
(Epoch 0 / 6) train acc: 0.174000; val_acc: 0.195000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.115000; val_acc: 0.089000
(Iteration 201 / 978) loss: inf
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.100000; val_acc: 0.112000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.132000; val_acc: 0.131000
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.078000; val_acc: 0.074000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.113000; val_acc: 0.120000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.112000; val_acc: 0.111000
(Iteration 1 / 1960) loss: 2.439473
(Epoch 0 / 4) train acc: 0.134000; val_acc: 0.151000
(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.077000; val_acc: 0.052000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf
(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.107000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.090000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
(Epoch 4 / 4) train acc: 0.086000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.444459
(Epoch 0 / 4) train acc: 0.166000; val_acc: 0.172000
(Iteration 101 / 980) loss: inf
```

```
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.096000; val_acc: 0.095000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
(Epoch 2 / 4) train acc: 0.117000; val_acc: 0.147000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
(Epoch 3 / 4) train acc: 0.105000; val_acc: 0.119000
(Iteration 801 / 980) loss: nan
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.109000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.452074
(Epoch 0 / 4) train acc: 0.217000; val_acc: 0.218000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.117000; val_acc: 0.096000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.077000; val_acc: 0.079000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.139000; val_acc: 0.144000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.099000; val_acc: 0.092000
(Iteration 1 / 1960) loss: 2.486521
(Epoch 0 / 4) train acc: 0.196000; val_acc: 0.209000
(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.120000; val_acc: 0.127000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf
(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.111000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.091000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
```

```
(Epoch 4 / 4) train acc: 0.098000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.484201
(Epoch 0 / 4) train acc: 0.184000; val_acc: 0.191000
(Iteration 101 / 980) loss: inf
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.110000; val_acc: 0.098000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
(Epoch 2 / 4) train acc: 0.131000; val_acc: 0.115000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
(Epoch 3 / 4) train acc: 0.071000; val_acc: 0.055000
(Iteration 801 / 980) loss: nan
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.101000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.493634
(Epoch 0 / 4) train acc: 0.221000; val_acc: 0.221000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.090000; val_acc: 0.113000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.057000; val_acc: 0.070000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.138000; val_acc: 0.133000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.168000; val_acc: 0.183000
(Iteration 1 / 1960) loss: 2.543081
(Epoch 0 / 4) train acc: 0.183000; val_acc: 0.183000
(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.075000; val_acc: 0.075000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf
(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.084000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.094000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
```

```
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
(Epoch 4 / 4) train acc: 0.097000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.534449
(Epoch 0 / 4) train acc: 0.187000; val_acc: 0.187000
(Iteration 101 / 980) loss: inf
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.142000; val_acc: 0.142000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
(Epoch 2 / 4) train acc: 0.087000; val_acc: 0.058000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
(Epoch 3 / 4) train acc: 0.116000; val_acc: 0.116000
(Iteration 801 / 980) loss: inf
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.089000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.533590
(Epoch 0 / 4) train acc: 0.201000; val_acc: 0.179000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.066000; val_acc: 0.062000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.092000; val_acc: 0.083000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.159000; val_acc: 0.155000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.146000; val_acc: 0.144000
(Iteration 1 / 1960) loss: 2.586115
(Epoch 0 / 4) train acc: 0.156000; val_acc: 0.135000
(Iteration 101 / 1960) loss: inf
(Iteration 201 / 1960) loss: inf
(Iteration 301 / 1960) loss: inf
(Iteration 401 / 1960) loss: inf
(Epoch 1 / 4) train acc: 0.087000; val_acc: 0.099000
(Iteration 501 / 1960) loss: inf
(Iteration 601 / 1960) loss: inf
(Iteration 701 / 1960) loss: inf
(Iteration 801 / 1960) loss: nan
(Iteration 901 / 1960) loss: nan
(Epoch 2 / 4) train acc: 0.103000; val_acc: 0.087000
(Iteration 1001 / 1960) loss: nan
(Iteration 1101 / 1960) loss: nan
(Iteration 1201 / 1960) loss: nan
```

```
(Iteration 1301 / 1960) loss: nan
(Iteration 1401 / 1960) loss: nan
(Epoch 3 / 4) train acc: 0.100000; val_acc: 0.087000
(Iteration 1501 / 1960) loss: nan
(Iteration 1601 / 1960) loss: nan
(Iteration 1701 / 1960) loss: nan
(Iteration 1801 / 1960) loss: nan
(Iteration 1901 / 1960) loss: nan
(Epoch 4 / 4) train acc: 0.115000; val_acc: 0.087000
(Iteration 1 / 980) loss: 2.583270
(Epoch 0 / 4) train acc: 0.174000; val_acc: 0.172000
(Iteration 101 / 980) loss: inf
(Iteration 201 / 980) loss: inf
(Epoch 1 / 4) train acc: 0.080000; val_acc: 0.086000
(Iteration 301 / 980) loss: inf
(Iteration 401 / 980) loss: inf
(Epoch 2 / 4) train acc: 0.116000; val_acc: 0.117000
(Iteration 501 / 980) loss: inf
(Iteration 601 / 980) loss: inf
(Iteration 701 / 980) loss: inf
(Epoch 3 / 4) train acc: 0.132000; val_acc: 0.135000
(Iteration 801 / 980) loss: inf
(Iteration 901 / 980) loss: nan
(Epoch 4 / 4) train acc: 0.090000; val_acc: 0.087000
(Iteration 1 / 652) loss: 2.577528
(Epoch 0 / 4) train acc: 0.216000; val_acc: 0.211000
(Iteration 101 / 652) loss: inf
(Epoch 1 / 4) train acc: 0.088000; val_acc: 0.080000
(Iteration 201 / 652) loss: inf
(Iteration 301 / 652) loss: inf
(Epoch 2 / 4) train acc: 0.088000; val_acc: 0.083000
(Iteration 401 / 652) loss: inf
(Epoch 3 / 4) train acc: 0.102000; val_acc: 0.092000
(Iteration 501 / 652) loss: inf
(Iteration 601 / 652) loss: inf
(Epoch 4 / 4) train acc: 0.146000; val_acc: 0.153000
(Iteration 1 / 2450) loss: 2.443612
(Epoch 0 / 5) train acc: 0.164000; val_acc: 0.158000
(Iteration 101 / 2450) loss: inf
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.078000; val_acc: 0.079000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
```

```
(Epoch 2 / 5) train acc: 0.089000; val_acc: 0.087000
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
(Iteration 1301 / 2450) loss: nan
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.107000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.094000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
(Epoch 5 / 5) train acc: 0.113000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.432786
(Epoch 0 / 5) train acc: 0.184000; val_acc: 0.182000
(Iteration 101 / 1225) loss: inf
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.106000; val_acc: 0.082000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.113000; val_acc: 0.098000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.073000; val_acc: 0.058000
(Iteration 801 / 1225) loss: inf
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.099000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.090000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.441143
(Epoch 0 / 5) train acc: 0.207000; val_acc: 0.218000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.101000; val_acc: 0.107000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.156000; val_acc: 0.137000
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.119000; val_acc: 0.085000
(Iteration 501 / 815) loss: inf
```

```
(Iteration 601 / 815) loss: inf
(Epoch 4 / 5) train acc: 0.091000; val_acc: 0.086000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
(Epoch 5 / 5) train acc: 0.090000; val_acc: 0.084000
(Iteration 1 / 2450) loss: 2.492145
(Epoch 0 / 5) train acc: 0.163000; val_acc: 0.160000
(Iteration 101 / 2450) loss: inf
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.086000; val_acc: 0.089000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
(Epoch 2 / 5) train acc: 0.108000; val_acc: 0.087000
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
(Iteration 1301 / 2450) loss: nan
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.109000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.095000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
(Epoch 5 / 5) train acc: 0.111000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.488500
(Epoch 0 / 5) train acc: 0.144000; val_acc: 0.165000
(Iteration 101 / 1225) loss: inf
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.127000; val_acc: 0.112000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.055000; val_acc: 0.061000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.090000; val_acc: 0.111000
```

```
(Iteration 801 / 1225) loss: inf
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.100000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.088000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.487695
(Epoch 0 / 5) train acc: 0.169000; val_acc: 0.157000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.110000; val_acc: 0.115000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.091000; val_acc: 0.107000
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.104000; val_acc: 0.091000
(Iteration 501 / 815) loss: inf
(Iteration 601 / 815) loss: inf
(Epoch 4 / 5) train acc: 0.127000; val_acc: 0.117000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
(Epoch 5 / 5) train acc: 0.116000; val_acc: 0.092000
(Iteration 1 / 2450) loss: 2.540551
(Epoch 0 / 5) train acc: 0.194000; val_acc: 0.201000
(Iteration 101 / 2450) loss: inf
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.079000; val_acc: 0.079000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
(Epoch 2 / 5) train acc: 0.103000; val_acc: 0.087000
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
(Iteration 1301 / 2450) loss: nan
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.110000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.120000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
```

```
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
(Epoch 5 / 5) train acc: 0.089000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.535023
(Epoch 0 / 5) train acc: 0.176000; val_acc: 0.196000
(Iteration 101 / 1225) loss: inf
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.102000; val_acc: 0.105000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.107000; val_acc: 0.099000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.142000; val_acc: 0.115000
(Iteration 801 / 1225) loss: inf
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.106000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.098000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.531685
(Epoch 0 / 5) train acc: 0.207000; val_acc: 0.212000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.094000; val_acc: 0.075000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.099000; val_acc: 0.107000
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.114000; val_acc: 0.104000
(Iteration 501 / 815) loss: inf
(Iteration 601 / 815) loss: inf
(Epoch 4 / 5) train acc: 0.138000; val_acc: 0.127000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
(Epoch 5 / 5) train acc: 0.109000; val_acc: 0.109000
(Iteration 1 / 2450) loss: 2.575394
(Epoch 0 / 5) train acc: 0.159000; val_acc: 0.155000
(Iteration 101 / 2450) loss: inf
(Iteration 201 / 2450) loss: inf
(Iteration 301 / 2450) loss: inf
(Iteration 401 / 2450) loss: inf
(Epoch 1 / 5) train acc: 0.163000; val_acc: 0.175000
(Iteration 501 / 2450) loss: inf
(Iteration 601 / 2450) loss: inf
```

```
(Iteration 701 / 2450) loss: inf
(Iteration 801 / 2450) loss: nan
(Iteration 901 / 2450) loss: nan
(Epoch 2 / 5) train acc: 0.097000; val_acc: 0.087000
(Iteration 1001 / 2450) loss: nan
(Iteration 1101 / 2450) loss: nan
(Iteration 1201 / 2450) loss: nan
(Iteration 1301 / 2450) loss: nan
(Iteration 1401 / 2450) loss: nan
(Epoch 3 / 5) train acc: 0.091000; val_acc: 0.087000
(Iteration 1501 / 2450) loss: nan
(Iteration 1601 / 2450) loss: nan
(Iteration 1701 / 2450) loss: nan
(Iteration 1801 / 2450) loss: nan
(Iteration 1901 / 2450) loss: nan
(Epoch 4 / 5) train acc: 0.093000; val_acc: 0.087000
(Iteration 2001 / 2450) loss: nan
(Iteration 2101 / 2450) loss: nan
(Iteration 2201 / 2450) loss: nan
(Iteration 2301 / 2450) loss: nan
(Iteration 2401 / 2450) loss: nan
(Epoch 5 / 5) train acc: 0.097000; val_acc: 0.087000
(Iteration 1 / 1225) loss: 2.583826
(Epoch 0 / 5) train acc: 0.181000; val_acc: 0.190000
(Iteration 101 / 1225) loss: inf
(Iteration 201 / 1225) loss: inf
(Epoch 1 / 5) train acc: 0.107000; val_acc: 0.098000
(Iteration 301 / 1225) loss: inf
(Iteration 401 / 1225) loss: inf
(Epoch 2 / 5) train acc: 0.080000; val_acc: 0.102000
(Iteration 501 / 1225) loss: inf
(Iteration 601 / 1225) loss: inf
(Iteration 701 / 1225) loss: inf
(Epoch 3 / 5) train acc: 0.127000; val_acc: 0.135000
(Iteration 801 / 1225) loss: inf
(Iteration 901 / 1225) loss: nan
(Epoch 4 / 5) train acc: 0.089000; val_acc: 0.087000
(Iteration 1001 / 1225) loss: nan
(Iteration 1101 / 1225) loss: nan
(Iteration 1201 / 1225) loss: nan
(Epoch 5 / 5) train acc: 0.086000; val_acc: 0.087000
(Iteration 1 / 815) loss: 2.583721
(Epoch 0 / 5) train acc: 0.161000; val_acc: 0.177000
(Iteration 101 / 815) loss: inf
(Epoch 1 / 5) train acc: 0.096000; val_acc: 0.122000
(Iteration 201 / 815) loss: inf
(Iteration 301 / 815) loss: inf
(Epoch 2 / 5) train acc: 0.126000; val_acc: 0.104000
```

```
(Iteration 401 / 815) loss: inf
(Epoch 3 / 5) train acc: 0.066000; val_acc: 0.068000
(Iteration 501 / 815) loss: inf
(Iteration 601 / 815) loss: inf
(Epoch 4 / 5) train acc: 0.072000; val_acc: 0.067000
(Iteration 701 / 815) loss: inf
(Iteration 801 / 815) loss: inf
(Epoch 5 / 5) train acc: 0.101000; val_acc: 0.109000
(Iteration 1 / 2940) loss: 2.440660
(Epoch 0 / 6) train acc: 0.153000; val_acc: 0.156000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.086000; val_acc: 0.097000
(Iteration 501 / 2940) loss: inf
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
(Epoch 2 / 6) train acc: 0.095000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.090000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.111000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.097000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
(Epoch 6 / 6) train acc: 0.104000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.442294
(Epoch 0 / 6) train acc: 0.193000; val_acc: 0.196000
(Iteration 101 / 1470) loss: inf
```

```
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.113000; val_acc: 0.094000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
(Epoch 2 / 6) train acc: 0.098000; val_acc: 0.092000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
(Epoch 3 / 6) train acc: 0.092000; val_acc: 0.109000
(Iteration 801 / 1470) loss: inf
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.111000; val_acc: 0.087000
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.095000; val_acc: 0.087000
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.107000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.441442
(Epoch 0 / 6) train acc: 0.168000; val_acc: 0.160000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.152000; val_acc: 0.158000
(Iteration 201 / 978) loss: inf
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.094000; val_acc: 0.119000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.073000; val_acc: 0.049000
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.114000; val_acc: 0.106000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.152000; val_acc: 0.144000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.117000; val_acc: 0.087000
(Iteration 1 / 2940) loss: 2.493937
(Epoch 0 / 6) train acc: 0.152000; val_acc: 0.133000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.055000; val_acc: 0.046000
(Iteration 501 / 2940) loss: inf
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
```

```
(Epoch 2 / 6) train acc: 0.100000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.112000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.105000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.099000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
(Epoch 6 / 6) train acc: 0.090000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.484555
(Epoch 0 / 6) train acc: 0.172000; val_acc: 0.205000
(Iteration 101 / 1470) loss: inf
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.080000; val_acc: 0.074000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
(Epoch 2 / 6) train acc: 0.105000; val_acc: 0.096000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
(Epoch 3 / 6) train acc: 0.089000; val_acc: 0.094000
(Iteration 801 / 1470) loss: inf
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.081000; val_acc: 0.087000
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.090000; val_acc: 0.087000
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.103000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.489611
```

```
(Epoch 0 / 6) train acc: 0.212000; val_acc: 0.230000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.097000; val_acc: 0.063000
(Iteration 201 / 978) loss: inf
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.106000; val_acc: 0.107000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.116000; val_acc: 0.107000
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.074000; val_acc: 0.096000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.094000; val_acc: 0.119000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.108000; val_acc: 0.112000
(Iteration 1 / 2940) loss: 2.543128
(Epoch 0 / 6) train acc: 0.114000; val_acc: 0.131000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.097000; val_acc: 0.100000
(Iteration 501 / 2940) loss: inf
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
(Epoch 2 / 6) train acc: 0.099000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.100000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.106000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.107000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
```

```
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
(Epoch 6 / 6) train acc: 0.102000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.538958
(Epoch 0 / 6) train acc: 0.146000; val_acc: 0.157000
(Iteration 101 / 1470) loss: inf
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.092000; val_acc: 0.082000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
(Epoch 2 / 6) train acc: 0.109000; val_acc: 0.076000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
(Epoch 3 / 6) train acc: 0.070000; val_acc: 0.083000
(Iteration 801 / 1470) loss: inf
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.111000; val_acc: 0.087000
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.098000; val_acc: 0.087000
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.096000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.527930
(Epoch 0 / 6) train acc: 0.181000; val_acc: 0.199000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.100000; val_acc: 0.122000
(Iteration 201 / 978) loss: inf
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.094000; val_acc: 0.083000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.079000; val_acc: 0.074000
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.128000; val_acc: 0.147000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.088000; val_acc: 0.071000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.102000; val_acc: 0.087000
(Iteration 1 / 2940) loss: 2.575453
(Epoch 0 / 6) train acc: 0.195000; val_acc: 0.193000
(Iteration 101 / 2940) loss: inf
(Iteration 201 / 2940) loss: inf
```

```
(Iteration 301 / 2940) loss: inf
(Iteration 401 / 2940) loss: inf
(Epoch 1 / 6) train acc: 0.079000; val_acc: 0.099000
(Iteration 501 / 2940) loss: inf
(Iteration 601 / 2940) loss: inf
(Iteration 701 / 2940) loss: inf
(Iteration 801 / 2940) loss: nan
(Iteration 901 / 2940) loss: nan
(Epoch 2 / 6) train acc: 0.092000; val_acc: 0.087000
(Iteration 1001 / 2940) loss: nan
(Iteration 1101 / 2940) loss: nan
(Iteration 1201 / 2940) loss: nan
(Iteration 1301 / 2940) loss: nan
(Iteration 1401 / 2940) loss: nan
(Epoch 3 / 6) train acc: 0.101000; val_acc: 0.087000
(Iteration 1501 / 2940) loss: nan
(Iteration 1601 / 2940) loss: nan
(Iteration 1701 / 2940) loss: nan
(Iteration 1801 / 2940) loss: nan
(Iteration 1901 / 2940) loss: nan
(Epoch 4 / 6) train acc: 0.100000; val_acc: 0.087000
(Iteration 2001 / 2940) loss: nan
(Iteration 2101 / 2940) loss: nan
(Iteration 2201 / 2940) loss: nan
(Iteration 2301 / 2940) loss: nan
(Iteration 2401 / 2940) loss: nan
(Epoch 5 / 6) train acc: 0.096000; val_acc: 0.087000
(Iteration 2501 / 2940) loss: nan
(Iteration 2601 / 2940) loss: nan
(Iteration 2701 / 2940) loss: nan
(Iteration 2801 / 2940) loss: nan
(Iteration 2901 / 2940) loss: nan
(Epoch 6 / 6) train acc: 0.098000; val_acc: 0.087000
(Iteration 1 / 1470) loss: 2.580258
(Epoch 0 / 6) train acc: 0.181000; val_acc: 0.188000
(Iteration 101 / 1470) loss: inf
(Iteration 201 / 1470) loss: inf
(Epoch 1 / 6) train acc: 0.121000; val_acc: 0.111000
(Iteration 301 / 1470) loss: inf
(Iteration 401 / 1470) loss: inf
(Epoch 2 / 6) train acc: 0.063000; val_acc: 0.072000
(Iteration 501 / 1470) loss: inf
(Iteration 601 / 1470) loss: inf
(Iteration 701 / 1470) loss: inf
(Epoch 3 / 6) train acc: 0.077000; val_acc: 0.085000
(Iteration 801 / 1470) loss: nan
(Iteration 901 / 1470) loss: nan
(Epoch 4 / 6) train acc: 0.101000; val_acc: 0.087000
```

```
(Iteration 1001 / 1470) loss: nan
(Iteration 1101 / 1470) loss: nan
(Iteration 1201 / 1470) loss: nan
(Epoch 5 / 6) train acc: 0.088000; val_acc: 0.087000
(Iteration 1301 / 1470) loss: nan
(Iteration 1401 / 1470) loss: nan
(Epoch 6 / 6) train acc: 0.103000; val_acc: 0.087000
(Iteration 1 / 978) loss: 2.582977
(Epoch 0 / 6) train acc: 0.206000; val_acc: 0.187000
(Iteration 101 / 978) loss: inf
(Epoch 1 / 6) train acc: 0.094000; val_acc: 0.103000
(Iteration 201 / 978) loss: inf
(Iteration 301 / 978) loss: inf
(Epoch 2 / 6) train acc: 0.101000; val_acc: 0.119000
(Iteration 401 / 978) loss: inf
(Epoch 3 / 6) train acc: 0.122000; val_acc: 0.153000
(Iteration 501 / 978) loss: inf
(Iteration 601 / 978) loss: inf
(Epoch 4 / 6) train acc: 0.153000; val_acc: 0.153000
(Iteration 701 / 978) loss: inf
(Iteration 801 / 978) loss: inf
(Epoch 5 / 6) train acc: 0.079000; val_acc: 0.080000
(Iteration 901 / 978) loss: inf
(Epoch 6 / 6) train acc: 0.074000; val_acc: 0.085000
(Iteration 1 / 1960) loss: 2.358155
(Epoch 0 / 4) train acc: 0.087000; val_acc: 0.092000
(Iteration 101 / 1960) loss: 2.277111
(Iteration 201 / 1960) loss: 2.127468
(Iteration 301 / 1960) loss: 2.052296
(Iteration 401 / 1960) loss: 1.982713
(Epoch 1 / 4) train acc: 0.283000; val_acc: 0.307000
(Iteration 501 / 1960) loss: 1.978434
(Iteration 601 / 1960) loss: 1.919517
(Iteration 701 / 1960) loss: 1.985635
(Iteration 801 / 1960) loss: 1.977296
(Iteration 901 / 1960) loss: 2.003084
(Epoch 2 / 4) train acc: 0.337000; val_acc: 0.373000
(Iteration 1001 / 1960) loss: 1.775870
(Iteration 1101 / 1960) loss: 1.878480
(Iteration 1201 / 1960) loss: 1.796209
(Iteration 1301 / 1960) loss: 1.623296
(Iteration 1401 / 1960) loss: 1.713073
(Epoch 3 / 4) train acc: 0.435000; val_acc: 0.401000
(Iteration 1501 / 1960) loss: 1.762253
(Iteration 1601 / 1960) loss: 1.608463
(Iteration 1701 / 1960) loss: 1.703089
(Iteration 1801 / 1960) loss: 1.762605
(Iteration 1901 / 1960) loss: 1.774457
```

```
(Epoch 4 / 4) train acc: 0.427000; val_acc: 0.427000
(Iteration 1 / 980) loss: 2.348783
(Epoch 0 / 4) train acc: 0.116000; val_acc: 0.152000
(Iteration 101 / 980) loss: 2.285481
(Iteration 201 / 980) loss: 2.089593
(Epoch 1 / 4) train acc: 0.273000; val_acc: 0.284000
(Iteration 301 / 980) loss: 2.112046
(Iteration 401 / 980) loss: 1.985868
(Epoch 2 / 4) train acc: 0.332000; val_acc: 0.313000
(Iteration 501 / 980) loss: 2.030917
(Iteration 601 / 980) loss: 1.968289
(Iteration 701 / 980) loss: 1.892189
(Epoch 3 / 4) train acc: 0.344000; val_acc: 0.349000
(Iteration 801 / 980) loss: 1.849656
(Iteration 901 / 980) loss: 1.869577
(Epoch 4 / 4) train acc: 0.333000; val_acc: 0.364000
(Iteration 1 / 652) loss: 2.348796
(Epoch 0 / 4) train acc: 0.082000; val_acc: 0.072000
(Iteration 101 / 652) loss: 2.261920
(Epoch 1 / 4) train acc: 0.219000; val_acc: 0.252000
(Iteration 201 / 652) loss: 2.141929
(Iteration 301 / 652) loss: 2.047580
(Epoch 2 / 4) train acc: 0.279000; val_acc: 0.286000
(Iteration 401 / 652) loss: 2.008397
(Epoch 3 / 4) train acc: 0.272000; val_acc: 0.308000
(Iteration 501 / 652) loss: 2.017498
(Iteration 601 / 652) loss: 1.927866
(Epoch 4 / 4) train acc: 0.313000; val_acc: 0.335000
(Iteration 1 / 1960) loss: 2.368345
(Epoch 0 / 4) train acc: 0.097000; val_acc: 0.087000
(Iteration 101 / 1960) loss: 2.318025
(Iteration 201 / 1960) loss: 2.180236
(Iteration 301 / 1960) loss: 1.973316
(Iteration 401 / 1960) loss: 2.014707
(Epoch 1 / 4) train acc: 0.315000; val_acc: 0.311000
(Iteration 501 / 1960) loss: 1.953665
(Iteration 601 / 1960) loss: 1.966841
(Iteration 701 / 1960) loss: 1.916323
(Iteration 801 / 1960) loss: 1.919898
(Iteration 901 / 1960) loss: 1.901932
(Epoch 2 / 4) train acc: 0.372000; val_acc: 0.370000
(Iteration 1001 / 1960) loss: 1.895135
(Iteration 1101 / 1960) loss: 1.819945
(Iteration 1201 / 1960) loss: 1.935146
(Iteration 1301 / 1960) loss: 1.775105
(Iteration 1401 / 1960) loss: 1.729290
(Epoch 3 / 4) train acc: 0.401000; val_acc: 0.407000
(Iteration 1501 / 1960) loss: 1.723579
```

```
(Iteration 1601 / 1960) loss: 1.719005
(Iteration 1701 / 1960) loss: 1.639719
(Iteration 1801 / 1960) loss: 1.770236
(Iteration 1901 / 1960) loss: 1.647082
(Epoch 4 / 4) train acc: 0.411000; val_acc: 0.434000
(Iteration 1 / 980) loss: 2.361696
(Epoch 0 / 4) train acc: 0.121000; val_acc: 0.112000
(Iteration 101 / 980) loss: 2.303355
(Iteration 201 / 980) loss: 2.200538
(Epoch 1 / 4) train acc: 0.263000; val_acc: 0.274000
(Iteration 301 / 980) loss: 2.185617
(Iteration 401 / 980) loss: 2.054453
(Epoch 2 / 4) train acc: 0.310000; val_acc: 0.315000
(Iteration 501 / 980) loss: 2.033114
(Iteration 601 / 980) loss: 1.987662
(Iteration 701 / 980) loss: 1.920909
(Epoch 3 / 4) train acc: 0.334000; val_acc: 0.342000
(Iteration 801 / 980) loss: 1.904921
(Iteration 901 / 980) loss: 1.806019
(Epoch 4 / 4) train acc: 0.371000; val_acc: 0.375000
(Iteration 1 / 652) loss: 2.367882
(Epoch 0 / 4) train acc: 0.099000; val_acc: 0.126000
(Iteration 101 / 652) loss: 2.311003
(Epoch 1 / 4) train acc: 0.242000; val_acc: 0.254000
(Iteration 201 / 652) loss: 2.161707
(Iteration 301 / 652) loss: 2.089223
(Epoch 2 / 4) train acc: 0.273000; val_acc: 0.283000
(Iteration 401 / 652) loss: 2.063426
(Epoch 3 / 4) train acc: 0.290000; val_acc: 0.313000
(Iteration 501 / 652) loss: 1.990723
(Iteration 601 / 652) loss: 1.908108
(Epoch 4 / 4) train acc: 0.327000; val_acc: 0.334000
(Iteration 1 / 1960) loss: 2.376256
(Epoch 0 / 4) train acc: 0.107000; val_acc: 0.099000
(Iteration 101 / 1960) loss: 2.295451
(Iteration 201 / 1960) loss: 2.196621
(Iteration 301 / 1960) loss: 2.100114
(Iteration 401 / 1960) loss: 2.099135
(Epoch 1 / 4) train acc: 0.315000; val_acc: 0.316000
(Iteration 501 / 1960) loss: 1.957266
(Iteration 601 / 1960) loss: 1.853722
(Iteration 701 / 1960) loss: 1.850979
(Iteration 801 / 1960) loss: 1.860694
(Iteration 901 / 1960) loss: 1.814243
(Epoch 2 / 4) train acc: 0.382000; val_acc: 0.387000
(Iteration 1001 / 1960) loss: 1.688128
(Iteration 1101 / 1960) loss: 1.930064
(Iteration 1201 / 1960) loss: 1.834731
```

```
(Iteration 1301 / 1960) loss: 1.874982
(Iteration 1401 / 1960) loss: 1.942644
(Epoch 3 / 4) train acc: 0.394000; val_acc: 0.407000
(Iteration 1501 / 1960) loss: 1.804983
(Iteration 1601 / 1960) loss: 1.649381
(Iteration 1701 / 1960) loss: 1.691154
(Iteration 1801 / 1960) loss: 1.704777
(Iteration 1901 / 1960) loss: 1.874501
(Epoch 4 / 4) train acc: 0.423000; val_acc: 0.430000
(Iteration 1 / 980) loss: 2.381744
(Epoch 0 / 4) train acc: 0.108000; val_acc: 0.120000
(Iteration 101 / 980) loss: 2.314259
(Iteration 201 / 980) loss: 2.179796
(Epoch 1 / 4) train acc: 0.255000; val_acc: 0.267000
(Iteration 301 / 980) loss: 2.182916
(Iteration 401 / 980) loss: 2.010648
(Epoch 2 / 4) train acc: 0.308000; val_acc: 0.311000
(Iteration 501 / 980) loss: 1.980143
(Iteration 601 / 980) loss: 1.988807
(Iteration 701 / 980) loss: 1.921711
(Epoch 3 / 4) train acc: 0.354000; val_acc: 0.340000
(Iteration 801 / 980) loss: 1.928733
(Iteration 901 / 980) loss: 1.845576
(Epoch 4 / 4) train acc: 0.353000; val_acc: 0.367000
(Iteration 1 / 652) loss: 2.380905
(Epoch 0 / 4) train acc: 0.096000; val_acc: 0.096000
(Iteration 101 / 652) loss: 2.307867
(Epoch 1 / 4) train acc: 0.252000; val_acc: 0.265000
(Iteration 201 / 652) loss: 2.240547
(Iteration 301 / 652) loss: 2.122282
(Epoch 2 / 4) train acc: 0.287000; val_acc: 0.288000
(Iteration 401 / 652) loss: 2.063597
(Epoch 3 / 4) train acc: 0.333000; val_acc: 0.307000
(Iteration 501 / 652) loss: 1.986598
(Iteration 601 / 652) loss: 1.931304
(Epoch 4 / 4) train acc: 0.342000; val_acc: 0.325000
(Iteration 1 / 1960) loss: 2.390965
(Epoch 0 / 4) train acc: 0.119000; val_acc: 0.103000
(Iteration 101 / 1960) loss: 2.341494
(Iteration 201 / 1960) loss: 2.227610
(Iteration 301 / 1960) loss: 2.164511
(Iteration 401 / 1960) loss: 2.062719
(Epoch 1 / 4) train acc: 0.295000; val_acc: 0.312000
(Iteration 501 / 1960) loss: 2.012039
(Iteration 601 / 1960) loss: 1.949553
(Iteration 701 / 1960) loss: 1.901690
(Iteration 801 / 1960) loss: 1.804121
(Iteration 901 / 1960) loss: 1.896953
```

```
(Epoch 2 / 4) train acc: 0.364000; val_acc: 0.371000
(Iteration 1001 / 1960) loss: 1.723053
(Iteration 1101 / 1960) loss: 1.831160
(Iteration 1201 / 1960) loss: 2.034979
(Iteration 1301 / 1960) loss: 1.820445
(Iteration 1401 / 1960) loss: 1.764227
(Epoch 3 / 4) train acc: 0.392000; val_acc: 0.406000
(Iteration 1501 / 1960) loss: 1.855296
(Iteration 1601 / 1960) loss: 1.859229
(Iteration 1701 / 1960) loss: 1.719577
(Iteration 1801 / 1960) loss: 1.816364
(Iteration 1901 / 1960) loss: 1.620411
(Epoch 4 / 4) train acc: 0.420000; val_acc: 0.430000
(Iteration 1 / 980) loss: 2.395921
(Epoch 0 / 4) train acc: 0.101000; val_acc: 0.116000
(Iteration 101 / 980) loss: 2.312990
(Iteration 201 / 980) loss: 2.235918
(Epoch 1 / 4) train acc: 0.254000; val_acc: 0.267000
(Iteration 301 / 980) loss: 2.108459
(Iteration 401 / 980) loss: 2.064468
(Epoch 2 / 4) train acc: 0.297000; val_acc: 0.310000
(Iteration 501 / 980) loss: 1.928899
(Iteration 601 / 980) loss: 1.964828
(Iteration 701 / 980) loss: 1.987398
(Epoch 3 / 4) train acc: 0.325000; val_acc: 0.335000
(Iteration 801 / 980) loss: 1.925870
(Iteration 901 / 980) loss: 1.822891
(Epoch 4 / 4) train acc: 0.370000; val_acc: 0.356000
(Iteration 1 / 652) loss: 2.393468
(Epoch 0 / 4) train acc: 0.105000; val_acc: 0.104000
(Iteration 101 / 652) loss: 2.337828
(Epoch 1 / 4) train acc: 0.211000; val_acc: 0.253000
(Iteration 201 / 652) loss: 2.228967
(Iteration 301 / 652) loss: 2.143891
(Epoch 2 / 4) train acc: 0.276000; val_acc: 0.281000
(Iteration 401 / 652) loss: 2.088939
(Epoch 3 / 4) train acc: 0.310000; val_acc: 0.304000
(Iteration 501 / 652) loss: 1.974205
(Iteration 601 / 652) loss: 2.017181
(Epoch 4 / 4) train acc: 0.306000; val_acc: 0.317000
(Iteration 1 / 2450) loss: 2.349366
(Epoch 0 / 5) train acc: 0.103000; val_acc: 0.108000
(Iteration 101 / 2450) loss: 2.281709
(Iteration 201 / 2450) loss: 2.215631
(Iteration 301 / 2450) loss: 2.077442
(Iteration 401 / 2450) loss: 1.989965
(Epoch 1 / 5) train acc: 0.342000; val_acc: 0.307000
(Iteration 501 / 2450) loss: 2.033408
```

```
(Iteration 601 / 2450) loss: 1.997617
(Iteration 701 / 2450) loss: 1.926796
(Iteration 801 / 2450) loss: 2.044274
(Iteration 901 / 2450) loss: 1.871688
(Epoch 2 / 5) train acc: 0.400000; val_acc: 0.364000
(Iteration 1001 / 2450) loss: 1.872708
(Iteration 1101 / 2450) loss: 1.752532
(Iteration 1201 / 2450) loss: 1.768931
(Iteration 1301 / 2450) loss: 1.877616
(Iteration 1401 / 2450) loss: 1.747213
(Epoch 3 / 5) train acc: 0.431000; val_acc: 0.404000
(Iteration 1501 / 2450) loss: 1.671635
(Iteration 1601 / 2450) loss: 1.775493
(Iteration 1701 / 2450) loss: 1.663531
(Iteration 1801 / 2450) loss: 1.750633
(Iteration 1901 / 2450) loss: 1.784428
(Epoch 4 / 5) train acc: 0.428000; val_acc: 0.436000
(Iteration 2001 / 2450) loss: 1.676046
(Iteration 2101 / 2450) loss: 1.646899
(Iteration 2201 / 2450) loss: 1.731495
(Iteration 2301 / 2450) loss: 1.711569
(Iteration 2401 / 2450) loss: 1.520608
(Epoch 5 / 5) train acc: 0.446000; val_acc: 0.442000
(Iteration 1 / 1225) loss: 2.346283
(Epoch 0 / 5) train acc: 0.114000; val_acc: 0.087000
(Iteration 101 / 1225) loss: 2.267237
(Iteration 201 / 1225) loss: 2.168432
(Epoch 1 / 5) train acc: 0.266000; val_acc: 0.277000
(Iteration 301 / 1225) loss: 2.031522
(Iteration 401 / 1225) loss: 2.028750
(Epoch 2 / 5) train acc: 0.328000; val_acc: 0.311000
(Iteration 501 / 1225) loss: 1.900210
(Iteration 601 / 1225) loss: 2.007740
(Iteration 701 / 1225) loss: 1.899475
(Epoch 3 / 5) train acc: 0.336000; val_acc: 0.347000
(Iteration 801 / 1225) loss: 1.842536
(Iteration 901 / 1225) loss: 1.891917
(Epoch 4 / 5) train acc: 0.350000; val_acc: 0.367000
(Iteration 1001 / 1225) loss: 1.855614
(Iteration 1101 / 1225) loss: 1.866517
(Iteration 1201 / 1225) loss: 1.732729
(Epoch 5 / 5) train acc: 0.370000; val_acc: 0.388000
(Iteration 1 / 815) loss: 2.348185
(Epoch 0 / 5) train acc: 0.113000; val_acc: 0.090000
(Iteration 101 / 815) loss: 2.280362
(Epoch 1 / 5) train acc: 0.219000; val_acc: 0.227000
(Iteration 201 / 815) loss: 2.185621
(Iteration 301 / 815) loss: 2.137327
```

```
(Epoch 2 / 5) train acc: 0.261000; val_acc: 0.283000
(Iteration 401 / 815) loss: 2.006736
(Epoch 3 / 5) train acc: 0.310000; val_acc: 0.307000
(Iteration 501 / 815) loss: 1.929937
(Iteration 601 / 815) loss: 1.915022
(Epoch 4 / 5) train acc: 0.342000; val_acc: 0.328000
(Iteration 701 / 815) loss: 1.946876
(Iteration 801 / 815) loss: 1.942930
(Epoch 5 / 5) train acc: 0.348000; val_acc: 0.342000
(Iteration 1 / 2450) loss: 2.366263
(Epoch 0 / 5) train acc: 0.101000; val_acc: 0.087000
(Iteration 101 / 2450) loss: 2.290704
(Iteration 201 / 2450) loss: 2.232829
(Iteration 301 / 2450) loss: 2.049383
(Iteration 401 / 2450) loss: 2.075532
(Epoch 1 / 5) train acc: 0.342000; val_acc: 0.307000
(Iteration 501 / 2450) loss: 1.949182
(Iteration 601 / 2450) loss: 1.938286
(Iteration 701 / 2450) loss: 1.846246
(Iteration 801 / 2450) loss: 1.859574
(Iteration 901 / 2450) loss: 1.990184
(Epoch 2 / 5) train acc: 0.372000; val_acc: 0.359000
(Iteration 1001 / 2450) loss: 1.740328
(Iteration 1101 / 2450) loss: 1.953405
(Iteration 1201 / 2450) loss: 1.659976
(Iteration 1301 / 2450) loss: 1.895663
(Iteration 1401 / 2450) loss: 1.728129
(Epoch 3 / 5) train acc: 0.413000; val_acc: 0.405000
(Iteration 1501 / 2450) loss: 1.903196
(Iteration 1601 / 2450) loss: 1.752147
(Iteration 1701 / 2450) loss: 1.744941
(Iteration 1801 / 2450) loss: 1.514394
(Iteration 1901 / 2450) loss: 1.661166
(Epoch 4 / 5) train acc: 0.433000; val_acc: 0.433000
(Iteration 2001 / 2450) loss: 1.882546
(Iteration 2101 / 2450) loss: 1.613810
(Iteration 2201 / 2450) loss: 1.621486
(Iteration 2301 / 2450) loss: 1.705743
(Iteration 2401 / 2450) loss: 1.562172
(Epoch 5 / 5) train acc: 0.429000; val_acc: 0.445000
(Iteration 1 / 1225) loss: 2.365466
(Epoch 0 / 5) train acc: 0.130000; val_acc: 0.127000
(Iteration 101 / 1225) loss: 2.288884
(Iteration 201 / 1225) loss: 2.204642
(Epoch 1 / 5) train acc: 0.273000; val_acc: 0.272000
(Iteration 301 / 1225) loss: 2.071799
(Iteration 401 / 1225) loss: 2.074847
(Epoch 2 / 5) train acc: 0.302000; val_acc: 0.319000
```

```
(Iteration 501 / 1225) loss: 1.978290
(Iteration 601 / 1225) loss: 2.037179
(Iteration 701 / 1225) loss: 1.940085
(Epoch 3 / 5) train acc: 0.330000; val_acc: 0.350000
(Iteration 801 / 1225) loss: 1.879154
(Iteration 901 / 1225) loss: 1.939147
(Epoch 4 / 5) train acc: 0.373000; val_acc: 0.366000
(Iteration 1001 / 1225) loss: 1.757287
(Iteration 1101 / 1225) loss: 1.787538
(Iteration 1201 / 1225) loss: 1.838013
(Epoch 5 / 5) train acc: 0.350000; val_acc: 0.379000
(Iteration 1 / 815) loss: 2.363882
(Epoch 0 / 5) train acc: 0.114000; val_acc: 0.128000
(Iteration 101 / 815) loss: 2.295804
(Epoch 1 / 5) train acc: 0.243000; val_acc: 0.262000
(Iteration 201 / 815) loss: 2.170260
(Iteration 301 / 815) loss: 2.112558
(Epoch 2 / 5) train acc: 0.283000; val_acc: 0.290000
(Iteration 401 / 815) loss: 2.007529
(Epoch 3 / 5) train acc: 0.312000; val_acc: 0.306000
(Iteration 501 / 815) loss: 1.987790
(Iteration 601 / 815) loss: 1.872239
(Epoch 4 / 5) train acc: 0.325000; val_acc: 0.331000
(Iteration 701 / 815) loss: 1.911300
(Iteration 801 / 815) loss: 1.894979
(Epoch 5 / 5) train acc: 0.354000; val_acc: 0.346000
(Iteration 1 / 2450) loss: 2.381954
(Epoch 0 / 5) train acc: 0.102000; val_acc: 0.109000
(Iteration 101 / 2450) loss: 2.314219
(Iteration 201 / 2450) loss: 2.180660
(Iteration 301 / 2450) loss: 2.196881
(Iteration 401 / 2450) loss: 2.110641
(Epoch 1 / 5) train acc: 0.317000; val_acc: 0.304000
(Iteration 501 / 2450) loss: 1.991263
(Iteration 601 / 2450) loss: 2.079330
(Iteration 701 / 2450) loss: 1.890255
(Iteration 801 / 2450) loss: 1.912260
(Iteration 901 / 2450) loss: 1.935829
(Epoch 2 / 5) train acc: 0.369000; val_acc: 0.378000
(Iteration 1001 / 2450) loss: 1.867899
(Iteration 1101 / 2450) loss: 1.729168
(Iteration 1201 / 2450) loss: 1.780307
(Iteration 1301 / 2450) loss: 1.727041
(Iteration 1401 / 2450) loss: 1.773910
(Epoch 3 / 5) train acc: 0.415000; val_acc: 0.400000
(Iteration 1501 / 2450) loss: 1.663979
(Iteration 1601 / 2450) loss: 1.696607
(Iteration 1701 / 2450) loss: 1.697262
```

```
(Iteration 1801 / 2450) loss: 1.788848
(Iteration 1901 / 2450) loss: 1.848958
(Epoch 4 / 5) train acc: 0.407000; val_acc: 0.429000
(Iteration 2001 / 2450) loss: 1.692292
(Iteration 2101 / 2450) loss: 1.752625
(Iteration 2201 / 2450) loss: 1.669089
(Iteration 2301 / 2450) loss: 1.459615
(Iteration 2401 / 2450) loss: 1.779955
(Epoch 5 / 5) train acc: 0.443000; val_acc: 0.448000
(Iteration 1 / 1225) loss: 2.379470
(Epoch 0 / 5) train acc: 0.094000; val_acc: 0.099000
(Iteration 101 / 1225) loss: 2.319276
(Iteration 201 / 1225) loss: 2.157909
(Epoch 1 / 5) train acc: 0.279000; val_acc: 0.273000
(Iteration 301 / 1225) loss: 2.147853
(Iteration 401 / 1225) loss: 2.062007
(Epoch 2 / 5) train acc: 0.320000; val_acc: 0.311000
(Iteration 501 / 1225) loss: 1.980913
(Iteration 601 / 1225) loss: 1.984008
(Iteration 701 / 1225) loss: 1.869840
(Epoch 3 / 5) train acc: 0.337000; val_acc: 0.339000
(Iteration 801 / 1225) loss: 1.910032
(Iteration 901 / 1225) loss: 1.852880
(Epoch 4 / 5) train acc: 0.356000; val_acc: 0.366000
(Iteration 1001 / 1225) loss: 1.886274
(Iteration 1101 / 1225) loss: 1.949887
(Iteration 1201 / 1225) loss: 1.821474
(Epoch 5 / 5) train acc: 0.395000; val_acc: 0.381000
(Iteration 1 / 815) loss: 2.375879
(Epoch 0 / 5) train acc: 0.108000; val_acc: 0.112000
(Iteration 101 / 815) loss: 2.296767
(Epoch 1 / 5) train acc: 0.264000; val_acc: 0.259000
(Iteration 201 / 815) loss: 2.196927
(Iteration 301 / 815) loss: 2.118928
(Epoch 2 / 5) train acc: 0.289000; val_acc: 0.288000
(Iteration 401 / 815) loss: 2.073037
(Epoch 3 / 5) train acc: 0.305000; val_acc: 0.306000
(Iteration 501 / 815) loss: 2.008241
(Iteration 601 / 815) loss: 1.993994
(Epoch 4 / 5) train acc: 0.335000; val_acc: 0.327000
(Iteration 701 / 815) loss: 1.949941
(Iteration 801 / 815) loss: 1.985705
(Epoch 5 / 5) train acc: 0.357000; val_acc: 0.348000
(Iteration 1 / 2450) loss: 2.390292
(Epoch 0 / 5) train acc: 0.111000; val_acc: 0.102000
(Iteration 101 / 2450) loss: 2.322263
(Iteration 201 / 2450) loss: 2.223616
(Iteration 301 / 2450) loss: 2.132405
```

```
(Iteration 401 / 2450) loss: 2.092007
(Epoch 1 / 5) train acc: 0.284000; val_acc: 0.303000
(Iteration 501 / 2450) loss: 2.092263
(Iteration 601 / 2450) loss: 1.910556
(Iteration 701 / 2450) loss: 1.940752
(Iteration 801 / 2450) loss: 1.891189
(Iteration 901 / 2450) loss: 1.968086
(Epoch 2 / 5) train acc: 0.354000; val_acc: 0.374000
(Iteration 1001 / 2450) loss: 1.741658
(Iteration 1101 / 2450) loss: 1.880962
(Iteration 1201 / 2450) loss: 1.927412
(Iteration 1301 / 2450) loss: 1.773301
(Iteration 1401 / 2450) loss: 1.866473
(Epoch 3 / 5) train acc: 0.393000; val_acc: 0.407000
(Iteration 1501 / 2450) loss: 1.733490
(Iteration 1601 / 2450) loss: 1.836352
(Iteration 1701 / 2450) loss: 1.775412
(Iteration 1801 / 2450) loss: 1.798256
(Iteration 1901 / 2450) loss: 1.713906
(Epoch 4 / 5) train acc: 0.411000; val_acc: 0.430000
(Iteration 2001 / 2450) loss: 1.896647
(Iteration 2101 / 2450) loss: 1.680121
(Iteration 2201 / 2450) loss: 1.895845
(Iteration 2301 / 2450) loss: 1.792344
(Iteration 2401 / 2450) loss: 1.606042
(Epoch 5 / 5) train acc: 0.460000; val_acc: 0.449000
(Iteration 1 / 1225) loss: 2.394762
(Epoch 0 / 5) train acc: 0.081000; val_acc: 0.079000
(Iteration 101 / 1225) loss: 2.327908
(Iteration 201 / 1225) loss: 2.181841
(Epoch 1 / 5) train acc: 0.271000; val_acc: 0.279000
(Iteration 301 / 1225) loss: 2.155254
(Iteration 401 / 1225) loss: 2.165811
(Epoch 2 / 5) train acc: 0.320000; val_acc: 0.308000
(Iteration 501 / 1225) loss: 1.969832
(Iteration 601 / 1225) loss: 2.011153
(Iteration 701 / 1225) loss: 2.005131
(Epoch 3 / 5) train acc: 0.357000; val_acc: 0.337000
(Iteration 801 / 1225) loss: 1.917216
(Iteration 901 / 1225) loss: 1.921576
(Epoch 4 / 5) train acc: 0.341000; val_acc: 0.370000
(Iteration 1001 / 1225) loss: 1.821694
(Iteration 1101 / 1225) loss: 1.818993
(Iteration 1201 / 1225) loss: 1.751365
(Epoch 5 / 5) train acc: 0.354000; val_acc: 0.382000
(Iteration 1 / 815) loss: 2.403171
(Epoch 0 / 5) train acc: 0.077000; val_acc: 0.074000
(Iteration 101 / 815) loss: 2.341746
```

```
(Epoch 1 / 5) train acc: 0.243000; val_acc: 0.263000
(Iteration 201 / 815) loss: 2.239515
(Iteration 301 / 815) loss: 2.116928
(Epoch 2 / 5) train acc: 0.277000; val_acc: 0.282000
(Iteration 401 / 815) loss: 2.126787
(Epoch 3 / 5) train acc: 0.310000; val_acc: 0.307000
(Iteration 501 / 815) loss: 1.993505
(Iteration 601 / 815) loss: 2.054687
(Epoch 4 / 5) train acc: 0.331000; val_acc: 0.323000
(Iteration 701 / 815) loss: 1.975697
(Iteration 801 / 815) loss: 2.021217
(Epoch 5 / 5) train acc: 0.350000; val_acc: 0.349000
(Iteration 1 / 2940) loss: 2.353617
(Epoch 0 / 6) train acc: 0.086000; val_acc: 0.094000
(Iteration 101 / 2940) loss: 2.267699
(Iteration 201 / 2940) loss: 2.178898
(Iteration 301 / 2940) loss: 2.094136
(Iteration 401 / 2940) loss: 2.037884
(Epoch 1 / 6) train acc: 0.318000; val_acc: 0.316000
(Iteration 501 / 2940) loss: 1.962224
(Iteration 601 / 2940) loss: 1.868856
(Iteration 701 / 2940) loss: 1.917560
(Iteration 801 / 2940) loss: 1.805073
(Iteration 901 / 2940) loss: 2.039030
(Epoch 2 / 6) train acc: 0.383000; val_acc: 0.380000
(Iteration 1001 / 2940) loss: 1.631674
(Iteration 1101 / 2940) loss: 1.865025
(Iteration 1201 / 2940) loss: 1.884691
(Iteration 1301 / 2940) loss: 1.856761
(Iteration 1401 / 2940) loss: 1.626547
(Epoch 3 / 6) train acc: 0.413000; val_acc: 0.398000
(Iteration 1501 / 2940) loss: 1.695827
(Iteration 1601 / 2940) loss: 1.544480
(Iteration 1701 / 2940) loss: 1.618268
(Iteration 1801 / 2940) loss: 1.828075
(Iteration 1901 / 2940) loss: 1.670632
(Epoch 4 / 6) train acc: 0.428000; val_acc: 0.419000
(Iteration 2001 / 2940) loss: 1.647825
(Iteration 2101 / 2940) loss: 1.729791
(Iteration 2201 / 2940) loss: 1.731150
(Iteration 2301 / 2940) loss: 1.705604
(Iteration 2401 / 2940) loss: 1.604631
(Epoch 5 / 6) train acc: 0.446000; val_acc: 0.439000
(Iteration 2501 / 2940) loss: 1.663549
(Iteration 2601 / 2940) loss: 1.636108
(Iteration 2701 / 2940) loss: 1.582771
(Iteration 2801 / 2940) loss: 1.435246
(Iteration 2901 / 2940) loss: 1.682021
```

```
(Epoch 6 / 6) train acc: 0.440000; val_acc: 0.444000
(Iteration 1 / 1470) loss: 2.351651
(Epoch 0 / 6) train acc: 0.077000; val_acc: 0.101000
(Iteration 101 / 1470) loss: 2.252626
(Iteration 201 / 1470) loss: 2.251180
(Epoch 1 / 6) train acc: 0.261000; val_acc: 0.276000
(Iteration 301 / 1470) loss: 2.123635
(Iteration 401 / 1470) loss: 2.028777
(Epoch 2 / 6) train acc: 0.322000; val_acc: 0.318000
(Iteration 501 / 1470) loss: 1.979642
(Iteration 601 / 1470) loss: 1.957117
(Iteration 701 / 1470) loss: 1.950777
(Epoch 3 / 6) train acc: 0.329000; val_acc: 0.348000
(Iteration 801 / 1470) loss: 1.888210
(Iteration 901 / 1470) loss: 1.918099
(Epoch 4 / 6) train acc: 0.358000; val_acc: 0.375000
(Iteration 1001 / 1470) loss: 1.831887
(Iteration 1101 / 1470) loss: 1.741625
(Iteration 1201 / 1470) loss: 1.772614
(Epoch 5 / 6) train acc: 0.369000; val_acc: 0.376000
(Iteration 1301 / 1470) loss: 1.803488
(Iteration 1401 / 1470) loss: 1.869505
(Epoch 6 / 6) train acc: 0.415000; val_acc: 0.389000
(Iteration 1 / 978) loss: 2.349327
(Epoch 0 / 6) train acc: 0.118000; val_acc: 0.106000
(Iteration 101 / 978) loss: 2.267820
(Epoch 1 / 6) train acc: 0.267000; val_acc: 0.256000
(Iteration 201 / 978) loss: 2.149610
(Iteration 301 / 978) loss: 2.090297
(Epoch 2 / 6) train acc: 0.265000; val_acc: 0.292000
(Iteration 401 / 978) loss: 1.995943
(Epoch 3 / 6) train acc: 0.311000; val_acc: 0.301000
(Iteration 501 / 978) loss: 2.016259
(Iteration 601 / 978) loss: 1.961928
(Epoch 4 / 6) train acc: 0.316000; val_acc: 0.327000
(Iteration 701 / 978) loss: 1.878492
(Iteration 801 / 978) loss: 1.870832
(Epoch 5 / 6) train acc: 0.329000; val_acc: 0.349000
(Iteration 901 / 978) loss: 1.820709
(Epoch 6 / 6) train acc: 0.377000; val_acc: 0.366000
(Iteration 1 / 2940) loss: 2.366839
(Epoch 0 / 6) train acc: 0.117000; val_acc: 0.112000
(Iteration 101 / 2940) loss: 2.286374
(Iteration 201 / 2940) loss: 2.116101
(Iteration 301 / 2940) loss: 2.118224
(Iteration 401 / 2940) loss: 2.216578
(Epoch 1 / 6) train acc: 0.289000; val_acc: 0.310000
(Iteration 501 / 2940) loss: 1.859093
```

```
(Iteration 601 / 2940) loss: 1.979580
(Iteration 701 / 2940) loss: 1.975182
(Iteration 801 / 2940) loss: 1.894433
(Iteration 901 / 2940) loss: 1.831246
(Epoch 2 / 6) train acc: 0.353000; val_acc: 0.375000
(Iteration 1001 / 2940) loss: 1.934915
(Iteration 1101 / 2940) loss: 2.014466
(Iteration 1201 / 2940) loss: 1.801396
(Iteration 1301 / 2940) loss: 1.801545
(Iteration 1401 / 2940) loss: 1.818758
(Epoch 3 / 6) train acc: 0.417000; val_acc: 0.406000
(Iteration 1501 / 2940) loss: 1.720159
(Iteration 1601 / 2940) loss: 1.829190
(Iteration 1701 / 2940) loss: 1.671369
(Iteration 1801 / 2940) loss: 1.667208
(Iteration 1901 / 2940) loss: 1.697446
(Epoch 4 / 6) train acc: 0.403000; val_acc: 0.430000
(Iteration 2001 / 2940) loss: 1.637168
(Iteration 2101 / 2940) loss: 1.637302
(Iteration 2201 / 2940) loss: 1.643829
(Iteration 2301 / 2940) loss: 1.653201
(Iteration 2401 / 2940) loss: 1.768922
(Epoch 5 / 6) train acc: 0.433000; val_acc: 0.441000
(Iteration 2501 / 2940) loss: 1.701442
(Iteration 2601 / 2940) loss: 1.668730
(Iteration 2701 / 2940) loss: 1.704361
(Iteration 2801 / 2940) loss: 1.668191
(Iteration 2901 / 2940) loss: 1.741107
(Epoch 6 / 6) train acc: 0.451000; val_acc: 0.445000
(Iteration 1 / 1470) loss: 2.362846
(Epoch 0 / 6) train acc: 0.104000; val_acc: 0.102000
(Iteration 101 / 1470) loss: 2.277612
(Iteration 201 / 1470) loss: 2.130855
(Epoch 1 / 6) train acc: 0.255000; val_acc: 0.282000
(Iteration 301 / 1470) loss: 2.076805
(Iteration 401 / 1470) loss: 2.039658
(Epoch 2 / 6) train acc: 0.309000; val_acc: 0.311000
(Iteration 501 / 1470) loss: 1.990853
(Iteration 601 / 1470) loss: 1.948272
(Iteration 701 / 1470) loss: 1.940939
(Epoch 3 / 6) train acc: 0.327000; val_acc: 0.342000
(Iteration 801 / 1470) loss: 1.840809
(Iteration 901 / 1470) loss: 1.787036
(Epoch 4 / 6) train acc: 0.357000; val_acc: 0.368000
(Iteration 1001 / 1470) loss: 1.861336
(Iteration 1101 / 1470) loss: 1.794912
(Iteration 1201 / 1470) loss: 1.756884
(Epoch 5 / 6) train acc: 0.390000; val_acc: 0.381000
```

```
(Iteration 1301 / 1470) loss: 1.753388
(Iteration 1401 / 1470) loss: 1.877617
(Epoch 6 / 6) train acc: 0.393000; val_acc: 0.395000
(Iteration 1 / 978) loss: 2.364202
(Epoch 0 / 6) train acc: 0.107000; val_acc: 0.119000
(Iteration 101 / 978) loss: 2.301901
(Epoch 1 / 6) train acc: 0.228000; val_acc: 0.240000
(Iteration 201 / 978) loss: 2.200899
(Iteration 301 / 978) loss: 2.089063
(Epoch 2 / 6) train acc: 0.280000; val_acc: 0.288000
(Iteration 401 / 978) loss: 2.051892
(Epoch 3 / 6) train acc: 0.296000; val_acc: 0.305000
(Iteration 501 / 978) loss: 2.003012
(Iteration 601 / 978) loss: 1.976361
(Epoch 4 / 6) train acc: 0.316000; val_acc: 0.336000
(Iteration 701 / 978) loss: 2.000690
(Iteration 801 / 978) loss: 1.878874
(Epoch 5 / 6) train acc: 0.355000; val_acc: 0.348000
(Iteration 901 / 978) loss: 1.890802
(Epoch 6 / 6) train acc: 0.341000; val_acc: 0.357000
(Iteration 1 / 2940) loss: 2.383019
(Epoch 0 / 6) train acc: 0.104000; val_acc: 0.100000
(Iteration 101 / 2940) loss: 2.317954
(Iteration 201 / 2940) loss: 2.165878
(Iteration 301 / 2940) loss: 2.134671
(Iteration 401 / 2940) loss: 2.122053
(Epoch 1 / 6) train acc: 0.302000; val_acc: 0.306000
(Iteration 501 / 2940) loss: 2.118247
(Iteration 601 / 2940) loss: 1.949400
(Iteration 701 / 2940) loss: 1.951711
(Iteration 801 / 2940) loss: 1.925358
(Iteration 901 / 2940) loss: 1.638454
(Epoch 2 / 6) train acc: 0.360000; val_acc: 0.377000
(Iteration 1001 / 2940) loss: 1.825261
(Iteration 1101 / 2940) loss: 1.873872
(Iteration 1201 / 2940) loss: 1.867367
(Iteration 1301 / 2940) loss: 1.725685
(Iteration 1401 / 2940) loss: 1.727747
(Epoch 3 / 6) train acc: 0.423000; val_acc: 0.401000
(Iteration 1501 / 2940) loss: 1.693883
(Iteration 1601 / 2940) loss: 1.766369
(Iteration 1701 / 2940) loss: 1.785905
(Iteration 1801 / 2940) loss: 1.639778
(Iteration 1901 / 2940) loss: 1.755290
(Epoch 4 / 6) train acc: 0.412000; val_acc: 0.421000
(Iteration 2001 / 2940) loss: 1.619894
(Iteration 2101 / 2940) loss: 1.748952
(Iteration 2201 / 2940) loss: 1.734898
```

```
(Iteration 2301 / 2940) loss: 1.806926
(Iteration 2401 / 2940) loss: 1.649944
(Epoch 5 / 6) train acc: 0.437000; val_acc: 0.436000
(Iteration 2501 / 2940) loss: 1.672140
(Iteration 2601 / 2940) loss: 1.699920
(Iteration 2701 / 2940) loss: 1.699635
(Iteration 2801 / 2940) loss: 1.681407
(Iteration 2901 / 2940) loss: 1.575450
(Epoch 6 / 6) train acc: 0.455000; val_acc: 0.449000
(Iteration 1 / 1470) loss: 2.380699
(Epoch 0 / 6) train acc: 0.101000; val_acc: 0.106000
(Iteration 101 / 1470) loss: 2.290577
(Iteration 201 / 1470) loss: 2.216087
(Epoch 1 / 6) train acc: 0.250000; val_acc: 0.258000
(Iteration 301 / 1470) loss: 2.055325
(Iteration 401 / 1470) loss: 2.039010
(Epoch 2 / 6) train acc: 0.304000; val_acc: 0.314000
(Iteration 501 / 1470) loss: 1.985099
(Iteration 601 / 1470) loss: 1.929341
(Iteration 701 / 1470) loss: 1.858334
(Epoch 3 / 6) train acc: 0.345000; val_acc: 0.342000
(Iteration 801 / 1470) loss: 1.946117
(Iteration 901 / 1470) loss: 1.848569
(Epoch 4 / 6) train acc: 0.346000; val_acc: 0.369000
(Iteration 1001 / 1470) loss: 1.835140
(Iteration 1101 / 1470) loss: 1.805474
(Iteration 1201 / 1470) loss: 1.881178
(Epoch 5 / 6) train acc: 0.402000; val_acc: 0.391000
(Iteration 1301 / 1470) loss: 1.817004
(Iteration 1401 / 1470) loss: 1.841961
(Epoch 6 / 6) train acc: 0.394000; val_acc: 0.398000
(Iteration 1 / 978) loss: 2.379606
(Epoch 0 / 6) train acc: 0.086000; val_acc: 0.091000
(Iteration 101 / 978) loss: 2.318909
(Epoch 1 / 6) train acc: 0.257000; val_acc: 0.252000
(Iteration 201 / 978) loss: 2.221551
(Iteration 301 / 978) loss: 2.168363
(Epoch 2 / 6) train acc: 0.275000; val_acc: 0.271000
(Iteration 401 / 978) loss: 2.078615
(Epoch 3 / 6) train acc: 0.316000; val_acc: 0.296000
(Iteration 501 / 978) loss: 1.967182
(Iteration 601 / 978) loss: 1.942672
(Epoch 4 / 6) train acc: 0.329000; val_acc: 0.322000
(Iteration 701 / 978) loss: 1.911171
(Iteration 801 / 978) loss: 1.959046
(Epoch 5 / 6) train acc: 0.341000; val_acc: 0.347000
(Iteration 901 / 978) loss: 1.881054
(Epoch 6 / 6) train acc: 0.362000; val_acc: 0.367000
```

```
(Iteration 1 / 2940) loss: 2.394384
(Epoch 0 / 6) train acc: 0.117000; val_acc: 0.113000
(Iteration 101 / 2940) loss: 2.328454
(Iteration 201 / 2940) loss: 2.218214
(Iteration 301 / 2940) loss: 2.068560
(Iteration 401 / 2940) loss: 2.157734
(Epoch 1 / 6) train acc: 0.318000; val_acc: 0.316000
(Iteration 501 / 2940) loss: 1.961614
(Iteration 601 / 2940) loss: 2.005784
(Iteration 701 / 2940) loss: 1.954134
(Iteration 801 / 2940) loss: 2.006010
(Iteration 901 / 2940) loss: 1.881643
(Epoch 2 / 6) train acc: 0.347000; val_acc: 0.372000
(Iteration 1001 / 2940) loss: 1.845146
(Iteration 1101 / 2940) loss: 1.785507
(Iteration 1201 / 2940) loss: 1.872008
(Iteration 1301 / 2940) loss: 1.961638
(Iteration 1401 / 2940) loss: 1.697940
(Epoch 3 / 6) train acc: 0.397000; val_acc: 0.420000
(Iteration 1501 / 2940) loss: 1.779240
(Iteration 1601 / 2940) loss: 1.597291
(Iteration 1701 / 2940) loss: 1.812804
(Iteration 1801 / 2940) loss: 1.676011
(Iteration 1901 / 2940) loss: 1.826681
(Epoch 4 / 6) train acc: 0.428000; val_acc: 0.424000
(Iteration 2001 / 2940) loss: 1.790584
(Iteration 2101 / 2940) loss: 1.866297
(Iteration 2201 / 2940) loss: 1.678269
(Iteration 2301 / 2940) loss: 1.553426
(Iteration 2401 / 2940) loss: 1.614021
(Epoch 5 / 6) train acc: 0.422000; val_acc: 0.444000
(Iteration 2501 / 2940) loss: 1.722880
(Iteration 2601 / 2940) loss: 1.735312
(Iteration 2701 / 2940) loss: 1.779562
(Iteration 2801 / 2940) loss: 1.845744
(Iteration 2901 / 2940) loss: 1.787040
(Epoch 6 / 6) train acc: 0.462000; val_acc: 0.450000
(Iteration 1 / 1470) loss: 2.393724
(Epoch 0 / 6) train acc: 0.107000; val_acc: 0.099000
(Iteration 101 / 1470) loss: 2.329712
(Iteration 201 / 1470) loss: 2.175949
(Epoch 1 / 6) train acc: 0.274000; val_acc: 0.268000
(Iteration 301 / 1470) loss: 2.117435
(Iteration 401 / 1470) loss: 2.068938
(Epoch 2 / 6) train acc: 0.301000; val_acc: 0.312000
(Iteration 501 / 1470) loss: 2.025304
(Iteration 601 / 1470) loss: 2.084373
(Iteration 701 / 1470) loss: 1.993948
```

```
(Epoch 3 / 6) train acc: 0.326000; val_acc: 0.349000
(Iteration 801 / 1470) loss: 1.897867
(Iteration 901 / 1470) loss: 1.900766
(Epoch 4 / 6) train acc: 0.358000; val_acc: 0.364000
(Iteration 1001 / 1470) loss: 2.001952
(Iteration 1101 / 1470) loss: 1.846944
(Iteration 1201 / 1470) loss: 1.872908
(Epoch 5 / 6) train acc: 0.381000; val_acc: 0.385000
(Iteration 1301 / 1470) loss: 1.748168
(Iteration 1401 / 1470) loss: 1.818887
(Epoch 6 / 6) train acc: 0.387000; val_acc: 0.394000
(Iteration 1 / 978) loss: 2.391801
(Epoch 0 / 6) train acc: 0.115000; val_acc: 0.145000
(Iteration 101 / 978) loss: 2.325527
(Epoch 1 / 6) train acc: 0.255000; val_acc: 0.285000
(Iteration 201 / 978) loss: 2.165266
(Iteration 301 / 978) loss: 2.137164
(Epoch 2 / 6) train acc: 0.287000; val_acc: 0.294000
(Iteration 401 / 978) loss: 2.027071
(Epoch 3 / 6) train acc: 0.302000; val_acc: 0.328000
(Iteration 501 / 978) loss: 1.989618
(Iteration 601 / 978) loss: 1.961214
(Epoch 4 / 6) train acc: 0.345000; val_acc: 0.345000
(Iteration 701 / 978) loss: 1.971738
(Iteration 801 / 978) loss: 1.894703
(Epoch 5 / 6) train acc: 0.332000; val_acc: 0.359000
(Iteration 901 / 978) loss: 1.892683
(Epoch 6 / 6) train acc: 0.359000; val_acc: 0.374000
(Iteration 1 / 1960) loss: 2.389976
(Epoch 0 / 4) train acc: 0.106000; val_acc: 0.111000
(Iteration 101 / 1960) loss: 2.293126
(Iteration 201 / 1960) loss: 2.114194
(Iteration 301 / 1960) loss: 2.087126
(Iteration 401 / 1960) loss: 2.063117
(Epoch 1 / 4) train acc: 0.304000; val_acc: 0.331000
(Iteration 501 / 1960) loss: 1.921974
(Iteration 601 / 1960) loss: 1.963270
(Iteration 701 / 1960) loss: 1.911918
(Iteration 801 / 1960) loss: 2.018687
(Iteration 901 / 1960) loss: 1.800693
(Epoch 2 / 4) train acc: 0.386000; val_acc: 0.384000
(Iteration 1001 / 1960) loss: 1.881747
(Iteration 1101 / 1960) loss: 1.767523
(Iteration 1201 / 1960) loss: 1.860754
(Iteration 1301 / 1960) loss: 1.685107
(Iteration 1401 / 1960) loss: 1.786799
(Epoch 3 / 4) train acc: 0.411000; val_acc: 0.409000
(Iteration 1501 / 1960) loss: 1.856578
```

```
(Iteration 1601 / 1960) loss: 1.723235
(Iteration 1701 / 1960) loss: 1.707381
(Iteration 1801 / 1960) loss: 1.815246
(Iteration 1901 / 1960) loss: 1.722526
(Epoch 4 / 4) train acc: 0.413000; val_acc: 0.429000
(Iteration 1 / 980) loss: 2.397232
(Epoch 0 / 4) train acc: 0.105000; val_acc: 0.111000
(Iteration 101 / 980) loss: 2.275658
(Iteration 201 / 980) loss: 2.163827
(Epoch 1 / 4) train acc: 0.290000; val_acc: 0.294000
(Iteration 301 / 980) loss: 2.049434
(Iteration 401 / 980) loss: 2.012839
(Epoch 2 / 4) train acc: 0.355000; val_acc: 0.327000
(Iteration 501 / 980) loss: 2.142391
(Iteration 601 / 980) loss: 1.963790
(Iteration 701 / 980) loss: 1.888531
(Epoch 3 / 4) train acc: 0.360000; val_acc: 0.362000
(Iteration 801 / 980) loss: 1.873093
(Iteration 901 / 980) loss: 1.796947
(Epoch 4 / 4) train acc: 0.379000; val_acc: 0.379000
(Iteration 1 / 652) loss: 2.393560
(Epoch 0 / 4) train acc: 0.098000; val_acc: 0.096000
(Iteration 101 / 652) loss: 2.308458
(Epoch 1 / 4) train acc: 0.250000; val_acc: 0.270000
(Iteration 201 / 652) loss: 2.183186
(Iteration 301 / 652) loss: 2.088700
(Epoch 2 / 4) train acc: 0.276000; val_acc: 0.301000
(Iteration 401 / 652) loss: 2.023604
(Epoch 3 / 4) train acc: 0.318000; val_acc: 0.321000
(Iteration 501 / 652) loss: 1.996795
(Iteration 601 / 652) loss: 2.001319
(Epoch 4 / 4) train acc: 0.358000; val_acc: 0.353000
(Iteration 1 / 1960) loss: 2.422821
(Epoch 0 / 4) train acc: 0.121000; val_acc: 0.139000
(Iteration 101 / 1960) loss: 2.280633
(Iteration 201 / 1960) loss: 2.235234
(Iteration 301 / 1960) loss: 2.069029
(Iteration 401 / 1960) loss: 1.965469
(Epoch 1 / 4) train acc: 0.320000; val_acc: 0.319000
(Iteration 501 / 1960) loss: 2.007420
(Iteration 601 / 1960) loss: 1.942319
(Iteration 701 / 1960) loss: 1.977923
(Iteration 801 / 1960) loss: 1.997678
(Iteration 901 / 1960) loss: 2.037673
(Epoch 2 / 4) train acc: 0.365000; val_acc: 0.388000
(Iteration 1001 / 1960) loss: 1.821960
(Iteration 1101 / 1960) loss: 1.780652
(Iteration 1201 / 1960) loss: 1.579777
```

```
(Iteration 1301 / 1960) loss: 1.771093
(Iteration 1401 / 1960) loss: 1.748374
(Epoch 3 / 4) train acc: 0.442000; val_acc: 0.412000
(Iteration 1501 / 1960) loss: 1.796549
(Iteration 1601 / 1960) loss: 1.848624
(Iteration 1701 / 1960) loss: 1.673261
(Iteration 1801 / 1960) loss: 1.880504
(Iteration 1901 / 1960) loss: 1.714562
(Epoch 4 / 4) train acc: 0.405000; val_acc: 0.444000
(Iteration 1 / 980) loss: 2.424388
(Epoch 0 / 4) train acc: 0.111000; val_acc: 0.102000
(Iteration 101 / 980) loss: 2.289419
(Iteration 201 / 980) loss: 2.218783
(Epoch 1 / 4) train acc: 0.290000; val_acc: 0.294000
(Iteration 301 / 980) loss: 2.069623
(Iteration 401 / 980) loss: 2.059156
(Epoch 2 / 4) train acc: 0.351000; val_acc: 0.334000
(Iteration 501 / 980) loss: 2.020258
(Iteration 601 / 980) loss: 1.969593
(Iteration 701 / 980) loss: 1.811248
(Epoch 3 / 4) train acc: 0.367000; val_acc: 0.364000
(Iteration 801 / 980) loss: 1.871788
(Iteration 901 / 980) loss: 1.863648
(Epoch 4 / 4) train acc: 0.372000; val_acc: 0.384000
(Iteration 1 / 652) loss: 2.424520
(Epoch 0 / 4) train acc: 0.096000; val_acc: 0.098000
(Iteration 101 / 652) loss: 2.312958
(Epoch 1 / 4) train acc: 0.264000; val_acc: 0.271000
(Iteration 201 / 652) loss: 2.195007
(Iteration 301 / 652) loss: 2.090252
(Epoch 2 / 4) train acc: 0.291000; val_acc: 0.305000
(Iteration 401 / 652) loss: 2.114732
(Epoch 3 / 4) train acc: 0.348000; val_acc: 0.327000
(Iteration 501 / 652) loss: 1.990492
(Iteration 601 / 652) loss: 1.963528
(Epoch 4 / 4) train acc: 0.353000; val_acc: 0.352000
(Iteration 1 / 1960) loss: 2.461301
(Epoch 0 / 4) train acc: 0.088000; val_acc: 0.096000
(Iteration 101 / 1960) loss: 2.362419
(Iteration 201 / 1960) loss: 2.252742
(Iteration 301 / 1960) loss: 2.059742
(Iteration 401 / 1960) loss: 2.094083
(Epoch 1 / 4) train acc: 0.343000; val_acc: 0.338000
(Iteration 501 / 1960) loss: 1.994132
(Iteration 601 / 1960) loss: 1.922105
(Iteration 701 / 1960) loss: 1.997491
(Iteration 801 / 1960) loss: 1.949202
(Iteration 901 / 1960) loss: 1.853179
```

```
(Epoch 2 / 4) train acc: 0.376000; val_acc: 0.398000
(Iteration 1001 / 1960) loss: 2.012399
(Iteration 1101 / 1960) loss: 2.028250
(Iteration 1201 / 1960) loss: 1.913847
(Iteration 1301 / 1960) loss: 1.787233
(Iteration 1401 / 1960) loss: 1.710552
(Epoch 3 / 4) train acc: 0.406000; val_acc: 0.415000
(Iteration 1501 / 1960) loss: 1.904932
(Iteration 1601 / 1960) loss: 1.968122
(Iteration 1701 / 1960) loss: 1.636699
(Iteration 1801 / 1960) loss: 1.799794
(Iteration 1901 / 1960) loss: 1.766724
(Epoch 4 / 4) train acc: 0.436000; val_acc: 0.439000
(Iteration 1 / 980) loss: 2.457030
(Epoch 0 / 4) train acc: 0.111000; val_acc: 0.084000
(Iteration 101 / 980) loss: 2.329294
(Iteration 201 / 980) loss: 2.235589
(Epoch 1 / 4) train acc: 0.268000; val_acc: 0.299000
(Iteration 301 / 980) loss: 2.089340
(Iteration 401 / 980) loss: 2.036242
(Epoch 2 / 4) train acc: 0.341000; val_acc: 0.333000
(Iteration 501 / 980) loss: 2.050947
(Iteration 601 / 980) loss: 2.011864
(Iteration 701 / 980) loss: 2.077295
(Epoch 3 / 4) train acc: 0.343000; val_acc: 0.361000
(Iteration 801 / 980) loss: 1.932202
(Iteration 901 / 980) loss: 2.020559
(Epoch 4 / 4) train acc: 0.352000; val_acc: 0.379000
(Iteration 1 / 652) loss: 2.452872
(Epoch 0 / 4) train acc: 0.135000; val_acc: 0.152000
(Iteration 101 / 652) loss: 2.323115
(Epoch 1 / 4) train acc: 0.273000; val_acc: 0.277000
(Iteration 201 / 652) loss: 2.239822
(Iteration 301 / 652) loss: 2.138257
(Epoch 2 / 4) train acc: 0.302000; val_acc: 0.302000
(Iteration 401 / 652) loss: 2.006699
(Epoch 3 / 4) train acc: 0.348000; val_acc: 0.328000
(Iteration 501 / 652) loss: 1.994710
(Iteration 601 / 652) loss: 2.048810
(Epoch 4 / 4) train acc: 0.355000; val_acc: 0.348000
(Iteration 1 / 1960) loss: 2.490278
(Epoch 0 / 4) train acc: 0.120000; val_acc: 0.100000
(Iteration 101 / 1960) loss: 2.371859
(Iteration 201 / 1960) loss: 2.282144
(Iteration 301 / 1960) loss: 2.156663
(Iteration 401 / 1960) loss: 2.051464
(Epoch 1 / 4) train acc: 0.328000; val_acc: 0.337000
(Iteration 501 / 1960) loss: 2.072258
```

```
(Iteration 601 / 1960) loss: 1.842250
(Iteration 701 / 1960) loss: 1.999007
(Iteration 801 / 1960) loss: 1.895642
(Iteration 901 / 1960) loss: 2.009892
(Epoch 2 / 4) train acc: 0.374000; val_acc: 0.402000
(Iteration 1001 / 1960) loss: 1.869030
(Iteration 1101 / 1960) loss: 1.993933
(Iteration 1201 / 1960) loss: 1.946378
(Iteration 1301 / 1960) loss: 1.806131
(Iteration 1401 / 1960) loss: 1.980688
(Epoch 3 / 4) train acc: 0.386000; val_acc: 0.416000
(Iteration 1501 / 1960) loss: 1.784677
(Iteration 1601 / 1960) loss: 1.718421
(Iteration 1701 / 1960) loss: 1.856824
(Iteration 1801 / 1960) loss: 1.762886
(Iteration 1901 / 1960) loss: 1.671657
(Epoch 4 / 4) train acc: 0.424000; val_acc: 0.440000
(Iteration 1 / 980) loss: 2.488657
(Epoch 0 / 4) train acc: 0.093000; val_acc: 0.087000
(Iteration 101 / 980) loss: 2.381040
(Iteration 201 / 980) loss: 2.280120
(Epoch 1 / 4) train acc: 0.262000; val_acc: 0.284000
(Iteration 301 / 980) loss: 2.155985
(Iteration 401 / 980) loss: 2.139159
(Epoch 2 / 4) train acc: 0.309000; val_acc: 0.321000
(Iteration 501 / 980) loss: 2.023167
(Iteration 601 / 980) loss: 2.162798
(Iteration 701 / 980) loss: 2.037463
(Epoch 3 / 4) train acc: 0.356000; val_acc: 0.349000
(Iteration 801 / 980) loss: 1.939423
(Iteration 901 / 980) loss: 1.918399
(Epoch 4 / 4) train acc: 0.376000; val_acc: 0.374000
(Iteration 1 / 652) loss: 2.487419
(Epoch 0 / 4) train acc: 0.074000; val_acc: 0.086000
(Iteration 101 / 652) loss: 2.365865
(Epoch 1 / 4) train acc: 0.236000; val_acc: 0.261000
(Iteration 201 / 652) loss: 2.311174
(Iteration 301 / 652) loss: 2.179963
(Epoch 2 / 4) train acc: 0.302000; val_acc: 0.302000
(Iteration 401 / 652) loss: 2.087918
(Epoch 3 / 4) train acc: 0.339000; val_acc: 0.324000
(Iteration 501 / 652) loss: 2.121418
(Iteration 601 / 652) loss: 2.003059
(Epoch 4 / 4) train acc: 0.387000; val_acc: 0.352000
(Iteration 1 / 2450) loss: 2.396078
(Epoch 0 / 5) train acc: 0.090000; val_acc: 0.091000
(Iteration 101 / 2450) loss: 2.287729
(Iteration 201 / 2450) loss: 2.182578
```

```
(Iteration 301 / 2450) loss: 1.975968
(Iteration 401 / 2450) loss: 2.038848
(Epoch 1 / 5) train acc: 0.334000; val_acc: 0.337000
(Iteration 501 / 2450) loss: 2.010648
(Iteration 601 / 2450) loss: 1.992655
(Iteration 701 / 2450) loss: 2.041162
(Iteration 801 / 2450) loss: 1.802365
(Iteration 901 / 2450) loss: 1.936274
(Epoch 2 / 5) train acc: 0.387000; val_acc: 0.382000
(Iteration 1001 / 2450) loss: 1.737121
(Iteration 1101 / 2450) loss: 1.818285
(Iteration 1201 / 2450) loss: 1.782515
(Iteration 1301 / 2450) loss: 1.589594
(Iteration 1401 / 2450) loss: 1.869588
(Epoch 3 / 5) train acc: 0.408000; val_acc: 0.409000
(Iteration 1501 / 2450) loss: 1.847494
(Iteration 1601 / 2450) loss: 1.725937
(Iteration 1701 / 2450) loss: 1.766254
(Iteration 1801 / 2450) loss: 1.720758
(Iteration 1901 / 2450) loss: 1.696036
(Epoch 4 / 5) train acc: 0.422000; val_acc: 0.443000
(Iteration 2001 / 2450) loss: 1.620002
(Iteration 2101 / 2450) loss: 1.672875
(Iteration 2201 / 2450) loss: 1.644342
(Iteration 2301 / 2450) loss: 1.701400
(Iteration 2401 / 2450) loss: 1.858842
(Epoch 5 / 5) train acc: 0.450000; val_acc: 0.444000
(Iteration 1 / 1225) loss: 2.394055
(Epoch 0 / 5) train acc: 0.093000; val_acc: 0.104000
(Iteration 101 / 1225) loss: 2.250355
(Iteration 201 / 1225) loss: 2.144258
(Epoch 1 / 5) train acc: 0.276000; val_acc: 0.294000
(Iteration 301 / 1225) loss: 2.008172
(Iteration 401 / 1225) loss: 2.024963
(Epoch 2 / 5) train acc: 0.321000; val_acc: 0.331000
(Iteration 501 / 1225) loss: 1.936040
(Iteration 601 / 1225) loss: 1.964717
(Iteration 701 / 1225) loss: 1.985374
(Epoch 3 / 5) train acc: 0.339000; val_acc: 0.365000
(Iteration 801 / 1225) loss: 1.959860
(Iteration 901 / 1225) loss: 1.895638
(Epoch 4 / 5) train acc: 0.377000; val_acc: 0.387000
(Iteration 1001 / 1225) loss: 1.834811
(Iteration 1101 / 1225) loss: 1.892652
(Iteration 1201 / 1225) loss: 1.870549
(Epoch 5 / 5) train acc: 0.383000; val_acc: 0.398000
(Iteration 1 / 815) loss: 2.399008
(Epoch 0 / 5) train acc: 0.088000; val_acc: 0.103000
```

```
(Iteration 101 / 815) loss: 2.279297
(Epoch 1 / 5) train acc: 0.268000; val_acc: 0.278000
(Iteration 201 / 815) loss: 2.180425
(Iteration 301 / 815) loss: 2.075658
(Epoch 2 / 5) train acc: 0.295000; val_acc: 0.316000
(Iteration 401 / 815) loss: 2.060945
(Epoch 3 / 5) train acc: 0.329000; val_acc: 0.329000
(Iteration 501 / 815) loss: 1.993569
(Iteration 601 / 815) loss: 1.945222
(Epoch 4 / 5) train acc: 0.373000; val_acc: 0.351000
(Iteration 701 / 815) loss: 1.962978
(Iteration 801 / 815) loss: 1.886671
(Epoch 5 / 5) train acc: 0.364000; val_acc: 0.372000
(Iteration 1 / 2450) loss: 2.428146
(Epoch 0 / 5) train acc: 0.096000; val_acc: 0.069000
(Iteration 101 / 2450) loss: 2.295982
(Iteration 201 / 2450) loss: 2.195937
(Iteration 301 / 2450) loss: 2.129495
(Iteration 401 / 2450) loss: 2.129698
(Epoch 1 / 5) train acc: 0.320000; val_acc: 0.331000
(Iteration 501 / 2450) loss: 2.117388
(Iteration 601 / 2450) loss: 1.992955
(Iteration 701 / 2450) loss: 1.918406
(Iteration 801 / 2450) loss: 1.806148
(Iteration 901 / 2450) loss: 1.869103
(Epoch 2 / 5) train acc: 0.373000; val_acc: 0.382000
(Iteration 1001 / 2450) loss: 1.843420
(Iteration 1101 / 2450) loss: 1.718342
(Iteration 1201 / 2450) loss: 1.937779
(Iteration 1301 / 2450) loss: 1.689843
(Iteration 1401 / 2450) loss: 1.877911
(Epoch 3 / 5) train acc: 0.422000; val_acc: 0.424000
(Iteration 1501 / 2450) loss: 1.781071
(Iteration 1601 / 2450) loss: 1.712997
(Iteration 1701 / 2450) loss: 1.650376
(Iteration 1801 / 2450) loss: 1.694927
(Iteration 1901 / 2450) loss: 1.557735
(Epoch 4 / 5) train acc: 0.426000; val_acc: 0.427000
(Iteration 2001 / 2450) loss: 1.743564
(Iteration 2101 / 2450) loss: 1.698544
(Iteration 2201 / 2450) loss: 1.542199
(Iteration 2301 / 2450) loss: 1.626951
(Iteration 2401 / 2450) loss: 1.818308
(Epoch 5 / 5) train acc: 0.425000; val_acc: 0.448000
(Iteration 1 / 1225) loss: 2.426610
(Epoch 0 / 5) train acc: 0.132000; val_acc: 0.142000
(Iteration 101 / 1225) loss: 2.297340
(Iteration 201 / 1225) loss: 2.216303
```

```
(Epoch 1 / 5) train acc: 0.296000; val_acc: 0.296000
(Iteration 301 / 1225) loss: 2.127263
(Iteration 401 / 1225) loss: 1.993950
(Epoch 2 / 5) train acc: 0.319000; val_acc: 0.332000
(Iteration 501 / 1225) loss: 2.000075
(Iteration 601 / 1225) loss: 1.922800
(Iteration 701 / 1225) loss: 1.987322
(Epoch 3 / 5) train acc: 0.351000; val_acc: 0.380000
(Iteration 801 / 1225) loss: 1.974356
(Iteration 901 / 1225) loss: 1.988188
(Epoch 4 / 5) train acc: 0.392000; val_acc: 0.384000
(Iteration 1001 / 1225) loss: 1.892378
(Iteration 1101 / 1225) loss: 1.883010
(Iteration 1201 / 1225) loss: 1.848557
(Epoch 5 / 5) train acc: 0.388000; val_acc: 0.400000
(Iteration 1 / 815) loss: 2.431842
(Epoch 0 / 5) train acc: 0.087000; val_acc: 0.074000
(Iteration 101 / 815) loss: 2.314083
(Epoch 1 / 5) train acc: 0.255000; val_acc: 0.266000
(Iteration 201 / 815) loss: 2.187551
(Iteration 301 / 815) loss: 2.105524
(Epoch 2 / 5) train acc: 0.295000; val_acc: 0.305000
(Iteration 401 / 815) loss: 2.070158
(Epoch 3 / 5) train acc: 0.316000; val_acc: 0.330000
(Iteration 501 / 815) loss: 1.997160
(Iteration 601 / 815) loss: 1.998756
(Epoch 4 / 5) train acc: 0.379000; val_acc: 0.362000
(Iteration 701 / 815) loss: 1.907576
(Iteration 801 / 815) loss: 1.898913
(Epoch 5 / 5) train acc: 0.347000; val_acc: 0.367000
(Iteration 1 / 2450) loss: 2.453363
(Epoch 0 / 5) train acc: 0.109000; val_acc: 0.105000
(Iteration 101 / 2450) loss: 2.327724
(Iteration 201 / 2450) loss: 2.145696
(Iteration 301 / 2450) loss: 2.135451
(Iteration 401 / 2450) loss: 2.060520
(Epoch 1 / 5) train acc: 0.332000; val_acc: 0.328000
(Iteration 501 / 2450) loss: 1.973258
(Iteration 601 / 2450) loss: 2.022660
(Iteration 701 / 2450) loss: 1.972343
(Iteration 801 / 2450) loss: 1.845288
(Iteration 901 / 2450) loss: 1.906047
(Epoch 2 / 5) train acc: 0.380000; val_acc: 0.388000
(Iteration 1001 / 2450) loss: 1.766925
(Iteration 1101 / 2450) loss: 1.854017
(Iteration 1201 / 2450) loss: 1.863446
(Iteration 1301 / 2450) loss: 1.953538
(Iteration 1401 / 2450) loss: 1.843623
```

```
(Epoch 3 / 5) train acc: 0.431000; val_acc: 0.422000
(Iteration 1501 / 2450) loss: 1.682495
(Iteration 1601 / 2450) loss: 1.869895
(Iteration 1701 / 2450) loss: 1.791528
(Iteration 1801 / 2450) loss: 1.707565
(Iteration 1901 / 2450) loss: 1.754722
(Epoch 4 / 5) train acc: 0.426000; val_acc: 0.424000
(Iteration 2001 / 2450) loss: 1.724871
(Iteration 2101 / 2450) loss: 1.737019
(Iteration 2201 / 2450) loss: 1.895958
(Iteration 2301 / 2450) loss: 1.747797
(Iteration 2401 / 2450) loss: 1.786358
(Epoch 5 / 5) train acc: 0.439000; val_acc: 0.451000
(Iteration 1 / 1225) loss: 2.457612
(Epoch 0 / 5) train acc: 0.094000; val_acc: 0.106000
(Iteration 101 / 1225) loss: 2.330608
(Iteration 201 / 1225) loss: 2.242427
(Epoch 1 / 5) train acc: 0.291000; val_acc: 0.300000
(Iteration 301 / 1225) loss: 2.086394
(Iteration 401 / 1225) loss: 2.060743
(Epoch 2 / 5) train acc: 0.349000; val_acc: 0.337000
(Iteration 501 / 1225) loss: 2.019328
(Iteration 601 / 1225) loss: 2.019755
(Iteration 701 / 1225) loss: 1.987617
(Epoch 3 / 5) train acc: 0.361000; val_acc: 0.373000
(Iteration 801 / 1225) loss: 2.010458
(Iteration 901 / 1225) loss: 1.996435
(Epoch 4 / 5) train acc: 0.387000; val_acc: 0.388000
(Iteration 1001 / 1225) loss: 1.949716
(Iteration 1101 / 1225) loss: 1.924103
(Iteration 1201 / 1225) loss: 1.821336
(Epoch 5 / 5) train acc: 0.381000; val_acc: 0.402000
(Iteration 1 / 815) loss: 2.458923
(Epoch 0 / 5) train acc: 0.102000; val_acc: 0.091000
(Iteration 101 / 815) loss: 2.358724
(Epoch 1 / 5) train acc: 0.263000; val_acc: 0.281000
(Iteration 201 / 815) loss: 2.201288
(Iteration 301 / 815) loss: 2.118971
(Epoch 2 / 5) train acc: 0.283000; val_acc: 0.306000
(Iteration 401 / 815) loss: 2.112152
(Epoch 3 / 5) train acc: 0.345000; val_acc: 0.325000
(Iteration 501 / 815) loss: 2.037040
(Iteration 601 / 815) loss: 2.049436
(Epoch 4 / 5) train acc: 0.345000; val_acc: 0.342000
(Iteration 701 / 815) loss: 2.001377
(Iteration 801 / 815) loss: 2.020041
(Epoch 5 / 5) train acc: 0.322000; val_acc: 0.370000
(Iteration 1 / 2450) loss: 2.490830
```

```
(Epoch 0 / 5) train acc: 0.095000; val_acc: 0.104000
(Iteration 101 / 2450) loss: 2.389522
(Iteration 201 / 2450) loss: 2.218879
(Iteration 301 / 2450) loss: 2.280510
(Iteration 401 / 2450) loss: 2.100423
(Epoch 1 / 5) train acc: 0.311000; val_acc: 0.314000
(Iteration 501 / 2450) loss: 2.169561
(Iteration 601 / 2450) loss: 1.944105
(Iteration 701 / 2450) loss: 1.853591
(Iteration 801 / 2450) loss: 2.002378
(Iteration 901 / 2450) loss: 1.917055
(Epoch 2 / 5) train acc: 0.401000; val_acc: 0.378000
(Iteration 1001 / 2450) loss: 1.917547
(Iteration 1101 / 2450) loss: 1.869559
(Iteration 1201 / 2450) loss: 1.834293
(Iteration 1301 / 2450) loss: 1.944970
(Iteration 1401 / 2450) loss: 1.737634
(Epoch 3 / 5) train acc: 0.426000; val_acc: 0.407000
(Iteration 1501 / 2450) loss: 1.767395
(Iteration 1601 / 2450) loss: 1.901305
(Iteration 1701 / 2450) loss: 1.866643
(Iteration 1801 / 2450) loss: 1.835867
(Iteration 1901 / 2450) loss: 1.772353
(Epoch 4 / 5) train acc: 0.418000; val_acc: 0.426000
(Iteration 2001 / 2450) loss: 1.810438
(Iteration 2101 / 2450) loss: 1.786016
(Iteration 2201 / 2450) loss: 1.859863
(Iteration 2301 / 2450) loss: 1.882911
(Iteration 2401 / 2450) loss: 1.756412
(Epoch 5 / 5) train acc: 0.454000; val_acc: 0.441000
(Iteration 1 / 1225) loss: 2.489466
(Epoch 0 / 5) train acc: 0.113000; val_acc: 0.124000
(Iteration 101 / 1225) loss: 2.369562
(Iteration 201 / 1225) loss: 2.286548
(Epoch 1 / 5) train acc: 0.274000; val_acc: 0.297000
(Iteration 301 / 1225) loss: 2.175304
(Iteration 401 / 1225) loss: 2.107718
(Epoch 2 / 5) train acc: 0.334000; val_acc: 0.333000
(Iteration 501 / 1225) loss: 2.041092
(Iteration 601 / 1225) loss: 1.946638
(Iteration 701 / 1225) loss: 1.957017
(Epoch 3 / 5) train acc: 0.345000; val_acc: 0.368000
(Iteration 801 / 1225) loss: 1.891442
(Iteration 901 / 1225) loss: 1.920725
(Epoch 4 / 5) train acc: 0.376000; val_acc: 0.386000
(Iteration 1001 / 1225) loss: 1.888672
(Iteration 1101 / 1225) loss: 1.829145
(Iteration 1201 / 1225) loss: 1.845041
```

```
(Epoch 5 / 5) train acc: 0.383000; val_acc: 0.395000
(Iteration 1 / 815) loss: 2.486536
(Epoch 0 / 5) train acc: 0.096000; val_acc: 0.075000
(Iteration 101 / 815) loss: 2.338999
(Epoch 1 / 5) train acc: 0.269000; val_acc: 0.273000
(Iteration 201 / 815) loss: 2.261950
(Iteration 301 / 815) loss: 2.181042
(Epoch 2 / 5) train acc: 0.330000; val_acc: 0.312000
(Iteration 401 / 815) loss: 2.137008
(Epoch 3 / 5) train acc: 0.310000; val_acc: 0.327000
(Iteration 501 / 815) loss: 2.099160
(Iteration 601 / 815) loss: 2.039527
(Epoch 4 / 5) train acc: 0.351000; val_acc: 0.346000
(Iteration 701 / 815) loss: 2.107690
(Iteration 801 / 815) loss: 2.053757
(Epoch 5 / 5) train acc: 0.360000; val_acc: 0.360000
(Iteration 1 / 2940) loss: 2.394520
(Epoch 0 / 6) train acc: 0.073000; val_acc: 0.089000
(Iteration 101 / 2940) loss: 2.292563
(Iteration 201 / 2940) loss: 2.192539
(Iteration 301 / 2940) loss: 2.154043
(Iteration 401 / 2940) loss: 2.033434
(Epoch 1 / 6) train acc: 0.331000; val_acc: 0.331000
(Iteration 501 / 2940) loss: 1.959433
(Iteration 601 / 2940) loss: 1.939550
(Iteration 701 / 2940) loss: 1.860504
(Iteration 801 / 2940) loss: 1.612471
(Iteration 901 / 2940) loss: 1.684302
(Epoch 2 / 6) train acc: 0.373000; val_acc: 0.379000
(Iteration 1001 / 2940) loss: 1.814925
(Iteration 1101 / 2940) loss: 1.834452
(Iteration 1201 / 2940) loss: 1.750092
(Iteration 1301 / 2940) loss: 1.770344
(Iteration 1401 / 2940) loss: 1.784162
(Epoch 3 / 6) train acc: 0.403000; val_acc: 0.418000
(Iteration 1501 / 2940) loss: 1.746683
(Iteration 1601 / 2940) loss: 1.727091
(Iteration 1701 / 2940) loss: 1.838051
(Iteration 1801 / 2940) loss: 1.640466
(Iteration 1901 / 2940) loss: 1.727714
(Epoch 4 / 6) train acc: 0.423000; val_acc: 0.437000
(Iteration 2001 / 2940) loss: 1.828130
(Iteration 2101 / 2940) loss: 1.849518
(Iteration 2201 / 2940) loss: 1.521556
(Iteration 2301 / 2940) loss: 1.802427
(Iteration 2401 / 2940) loss: 1.669535
(Epoch 5 / 6) train acc: 0.454000; val_acc: 0.449000
(Iteration 2501 / 2940) loss: 1.733565
```

```
(Iteration 2601 / 2940) loss: 1.693610
(Iteration 2701 / 2940) loss: 1.727647
(Iteration 2801 / 2940) loss: 1.635641
(Iteration 2901 / 2940) loss: 1.631470
(Epoch 6 / 6) train acc: 0.466000; val_acc: 0.460000
(Iteration 1 / 1470) loss: 2.397714
(Epoch 0 / 6) train acc: 0.103000; val_acc: 0.103000
(Iteration 101 / 1470) loss: 2.287853
(Iteration 201 / 1470) loss: 2.135966
(Epoch 1 / 6) train acc: 0.276000; val_acc: 0.298000
(Iteration 301 / 1470) loss: 2.128778
(Iteration 401 / 1470) loss: 2.018068
(Epoch 2 / 6) train acc: 0.359000; val_acc: 0.335000
(Iteration 501 / 1470) loss: 1.935553
(Iteration 601 / 1470) loss: 1.901756
(Iteration 701 / 1470) loss: 1.866913
(Epoch 3 / 6) train acc: 0.349000; val_acc: 0.372000
(Iteration 801 / 1470) loss: 1.863786
(Iteration 901 / 1470) loss: 1.952506
(Epoch 4 / 6) train acc: 0.368000; val_acc: 0.388000
(Iteration 1001 / 1470) loss: 1.834275
(Iteration 1101 / 1470) loss: 1.841132
(Iteration 1201 / 1470) loss: 1.883607
(Epoch 5 / 6) train acc: 0.382000; val_acc: 0.410000
(Iteration 1301 / 1470) loss: 1.670906
(Iteration 1401 / 1470) loss: 1.850696
(Epoch 6 / 6) train acc: 0.427000; val_acc: 0.419000
(Iteration 1 / 978) loss: 2.396902
(Epoch 0 / 6) train acc: 0.084000; val_acc: 0.093000
(Iteration 101 / 978) loss: 2.310583
(Epoch 1 / 6) train acc: 0.288000; val_acc: 0.272000
(Iteration 201 / 978) loss: 2.197187
(Iteration 301 / 978) loss: 2.123965
(Epoch 2 / 6) train acc: 0.296000; val_acc: 0.301000
(Iteration 401 / 978) loss: 2.008775
(Epoch 3 / 6) train acc: 0.331000; val_acc: 0.331000
(Iteration 501 / 978) loss: 1.982289
(Iteration 601 / 978) loss: 1.900066
(Epoch 4 / 6) train acc: 0.371000; val_acc: 0.342000
(Iteration 701 / 978) loss: 1.964390
(Iteration 801 / 978) loss: 1.923363
(Epoch 5 / 6) train acc: 0.369000; val_acc: 0.363000
(Iteration 901 / 978) loss: 1.875140
(Epoch 6 / 6) train acc: 0.376000; val_acc: 0.389000
(Iteration 1 / 2940) loss: 2.430024
(Epoch 0 / 6) train acc: 0.101000; val_acc: 0.096000
(Iteration 101 / 2940) loss: 2.313334
(Iteration 201 / 2940) loss: 2.218408
```

```
(Iteration 301 / 2940) loss: 2.133663
(Iteration 401 / 2940) loss: 2.018281
(Epoch 1 / 6) train acc: 0.320000; val_acc: 0.333000
(Iteration 501 / 2940) loss: 1.974937
(Iteration 601 / 2940) loss: 1.966080
(Iteration 701 / 2940) loss: 1.933924
(Iteration 801 / 2940) loss: 1.837163
(Iteration 901 / 2940) loss: 1.976029
(Epoch 2 / 6) train acc: 0.376000; val_acc: 0.387000
(Iteration 1001 / 2940) loss: 2.009929
(Iteration 1101 / 2940) loss: 1.718146
(Iteration 1201 / 2940) loss: 1.830519
(Iteration 1301 / 2940) loss: 1.886753
(Iteration 1401 / 2940) loss: 1.879559
(Epoch 3 / 6) train acc: 0.392000; val_acc: 0.429000
(Iteration 1501 / 2940) loss: 1.902984
(Iteration 1601 / 2940) loss: 2.011072
(Iteration 1701 / 2940) loss: 1.699649
(Iteration 1801 / 2940) loss: 1.800356
(Iteration 1901 / 2940) loss: 1.825983
(Epoch 4 / 6) train acc: 0.415000; val_acc: 0.430000
(Iteration 2001 / 2940) loss: 1.782704
(Iteration 2101 / 2940) loss: 1.631149
(Iteration 2201 / 2940) loss: 1.637785
(Iteration 2301 / 2940) loss: 1.790399
(Iteration 2401 / 2940) loss: 1.857589
(Epoch 5 / 6) train acc: 0.428000; val_acc: 0.444000
(Iteration 2501 / 2940) loss: 1.764003
(Iteration 2601 / 2940) loss: 1.516744
(Iteration 2701 / 2940) loss: 1.501475
(Iteration 2801 / 2940) loss: 1.806406
(Iteration 2901 / 2940) loss: 1.749236
(Epoch 6 / 6) train acc: 0.482000; val_acc: 0.454000
(Iteration 1 / 1470) loss: 2.422314
(Epoch 0 / 6) train acc: 0.104000; val_acc: 0.088000
(Iteration 101 / 1470) loss: 2.330227
(Iteration 201 / 1470) loss: 2.230081
(Epoch 1 / 6) train acc: 0.279000; val_acc: 0.292000
(Iteration 301 / 1470) loss: 2.127673
(Iteration 401 / 1470) loss: 2.025189
(Epoch 2 / 6) train acc: 0.303000; val_acc: 0.330000
(Iteration 501 / 1470) loss: 2.019530
(Iteration 601 / 1470) loss: 1.963279
(Iteration 701 / 1470) loss: 1.990652
(Epoch 3 / 6) train acc: 0.335000; val_acc: 0.363000
(Iteration 801 / 1470) loss: 1.971028
(Iteration 901 / 1470) loss: 1.807149
(Epoch 4 / 6) train acc: 0.392000; val_acc: 0.389000
```

```
(Iteration 1001 / 1470) loss: 1.919816
(Iteration 1101 / 1470) loss: 1.907685
(Iteration 1201 / 1470) loss: 1.822049
(Epoch 5 / 6) train acc: 0.410000; val_acc: 0.404000
(Iteration 1301 / 1470) loss: 1.754707
(Iteration 1401 / 1470) loss: 1.859108
(Epoch 6 / 6) train acc: 0.417000; val_acc: 0.410000
(Iteration 1 / 978) loss: 2.427699
(Epoch 0 / 6) train acc: 0.104000; val_acc: 0.084000
(Iteration 101 / 978) loss: 2.311615
(Epoch 1 / 6) train acc: 0.248000; val_acc: 0.279000
(Iteration 201 / 978) loss: 2.180008
(Iteration 301 / 978) loss: 2.073508
(Epoch 2 / 6) train acc: 0.318000; val_acc: 0.312000
(Iteration 401 / 978) loss: 2.044743
(Epoch 3 / 6) train acc: 0.354000; val_acc: 0.330000
(Iteration 501 / 978) loss: 2.088203
(Iteration 601 / 978) loss: 2.022681
(Epoch 4 / 6) train acc: 0.337000; val_acc: 0.342000
(Iteration 701 / 978) loss: 1.932213
(Iteration 801 / 978) loss: 1.980599
(Epoch 5 / 6) train acc: 0.358000; val_acc: 0.367000
(Iteration 901 / 978) loss: 1.943783
(Epoch 6 / 6) train acc: 0.347000; val_acc: 0.373000
(Iteration 1 / 2940) loss: 2.451161
(Epoch 0 / 6) train acc: 0.131000; val_acc: 0.119000
(Iteration 101 / 2940) loss: 2.379113
(Iteration 201 / 2940) loss: 2.190573
(Iteration 301 / 2940) loss: 2.118583
(Iteration 401 / 2940) loss: 2.102032
(Epoch 1 / 6) train acc: 0.331000; val_acc: 0.321000
(Iteration 501 / 2940) loss: 2.160921
(Iteration 601 / 2940) loss: 1.955118
(Iteration 701 / 2940) loss: 1.938464
(Iteration 801 / 2940) loss: 1.896030
(Iteration 901 / 2940) loss: 1.880467
(Epoch 2 / 6) train acc: 0.399000; val_acc: 0.385000
(Iteration 1001 / 2940) loss: 1.757480
(Iteration 1101 / 2940) loss: 1.859611
(Iteration 1201 / 2940) loss: 1.758433
(Iteration 1301 / 2940) loss: 1.794107
(Iteration 1401 / 2940) loss: 1.850758
(Epoch 3 / 6) train acc: 0.439000; val_acc: 0.422000
(Iteration 1501 / 2940) loss: 1.826490
(Iteration 1601 / 2940) loss: 1.876114
(Iteration 1701 / 2940) loss: 1.663682
(Iteration 1801 / 2940) loss: 1.822664
(Iteration 1901 / 2940) loss: 1.757659
```

```
(Epoch 4 / 6) train acc: 0.435000; val_acc: 0.432000
(Iteration 2001 / 2940) loss: 1.732337
(Iteration 2101 / 2940) loss: 1.890123
(Iteration 2201 / 2940) loss: 1.870724
(Iteration 2301 / 2940) loss: 1.748671
(Iteration 2401 / 2940) loss: 1.835897
(Epoch 5 / 6) train acc: 0.451000; val_acc: 0.456000
(Iteration 2501 / 2940) loss: 1.652973
(Iteration 2601 / 2940) loss: 1.638788
(Iteration 2701 / 2940) loss: 1.799654
(Iteration 2801 / 2940) loss: 1.810823
(Iteration 2901 / 2940) loss: 1.768561
(Epoch 6 / 6) train acc: 0.434000; val_acc: 0.449000
(Iteration 1 / 1470) loss: 2.460466
(Epoch 0 / 6) train acc: 0.091000; val_acc: 0.095000
(Iteration 101 / 1470) loss: 2.382436
(Iteration 201 / 1470) loss: 2.251545
(Epoch 1 / 6) train acc: 0.282000; val_acc: 0.303000
(Iteration 301 / 1470) loss: 2.138622
(Iteration 401 / 1470) loss: 2.103086
(Epoch 2 / 6) train acc: 0.337000; val_acc: 0.327000
(Iteration 501 / 1470) loss: 2.070512
(Iteration 601 / 1470) loss: 2.047846
(Iteration 701 / 1470) loss: 2.067507
(Epoch 3 / 6) train acc: 0.340000; val_acc: 0.362000
(Iteration 801 / 1470) loss: 1.965703
(Iteration 901 / 1470) loss: 1.911073
(Epoch 4 / 6) train acc: 0.329000; val_acc: 0.387000
(Iteration 1001 / 1470) loss: 1.949947
(Iteration 1101 / 1470) loss: 1.878436
(Iteration 1201 / 1470) loss: 1.885103
(Epoch 5 / 6) train acc: 0.406000; val_acc: 0.391000
(Iteration 1301 / 1470) loss: 1.815920
(Iteration 1401 / 1470) loss: 1.927280
(Epoch 6 / 6) train acc: 0.406000; val_acc: 0.406000
(Iteration 1 / 978) loss: 2.460419
(Epoch 0 / 6) train acc: 0.095000; val_acc: 0.099000
(Iteration 101 / 978) loss: 2.348822
(Epoch 1 / 6) train acc: 0.286000; val_acc: 0.272000
(Iteration 201 / 978) loss: 2.245261
(Iteration 301 / 978) loss: 2.171764
(Epoch 2 / 6) train acc: 0.325000; val_acc: 0.313000
(Iteration 401 / 978) loss: 2.052693
(Epoch 3 / 6) train acc: 0.358000; val_acc: 0.325000
(Iteration 501 / 978) loss: 2.052078
(Iteration 601 / 978) loss: 2.029820
(Epoch 4 / 6) train acc: 0.345000; val_acc: 0.345000
(Iteration 701 / 978) loss: 1.958281
```

```
(Iteration 801 / 978) loss: 2.009840
(Epoch 5 / 6) train acc: 0.348000; val_acc: 0.360000
(Iteration 901 / 978) loss: 1.975832
(Epoch 6 / 6) train acc: 0.379000; val_acc: 0.384000
(Iteration 1 / 2940) loss: 2.485840
(Epoch 0 / 6) train acc: 0.108000; val_acc: 0.091000
(Iteration 101 / 2940) loss: 2.375842
(Iteration 201 / 2940) loss: 2.202138
(Iteration 301 / 2940) loss: 2.101866
(Iteration 401 / 2940) loss: 2.043797
(Epoch 1 / 6) train acc: 0.325000; val_acc: 0.327000
(Iteration 501 / 2940) loss: 2.042542
(Iteration 601 / 2940) loss: 1.964243
(Iteration 701 / 2940) loss: 1.955624
(Iteration 801 / 2940) loss: 1.977954
(Iteration 901 / 2940) loss: 1.919747
(Epoch 2 / 6) train acc: 0.398000; val_acc: 0.382000
(Iteration 1001 / 2940) loss: 1.956422
(Iteration 1101 / 2940) loss: 1.805249
(Iteration 1201 / 2940) loss: 1.919454
(Iteration 1301 / 2940) loss: 1.869210
(Iteration 1401 / 2940) loss: 1.924350
(Epoch 3 / 6) train acc: 0.413000; val_acc: 0.413000
(Iteration 1501 / 2940) loss: 1.828476
(Iteration 1601 / 2940) loss: 1.809839
(Iteration 1701 / 2940) loss: 1.844996
(Iteration 1801 / 2940) loss: 1.718235
(Iteration 1901 / 2940) loss: 1.713030
(Epoch 4 / 6) train acc: 0.424000; val_acc: 0.444000
(Iteration 2001 / 2940) loss: 1.735125
(Iteration 2101 / 2940) loss: 1.649051
(Iteration 2201 / 2940) loss: 1.854823
(Iteration 2301 / 2940) loss: 1.930287
(Iteration 2401 / 2940) loss: 1.778931
(Epoch 5 / 6) train acc: 0.456000; val_acc: 0.450000
(Iteration 2501 / 2940) loss: 1.795473
(Iteration 2601 / 2940) loss: 1.670908
(Iteration 2701 / 2940) loss: 1.865750
(Iteration 2801 / 2940) loss: 1.702245
(Iteration 2901 / 2940) loss: 1.613452
(Epoch 6 / 6) train acc: 0.440000; val_acc: 0.453000
(Iteration 1 / 1470) loss: 2.489308
(Epoch 0 / 6) train acc: 0.084000; val_acc: 0.085000
(Iteration 101 / 1470) loss: 2.376150
(Iteration 201 / 1470) loss: 2.277031
(Epoch 1 / 6) train acc: 0.269000; val_acc: 0.285000
(Iteration 301 / 1470) loss: 2.188653
(Iteration 401 / 1470) loss: 2.204513
```

```
(Epoch 2 / 6) train acc: 0.310000; val_acc: 0.337000
(Iteration 501 / 1470) loss: 2.002898
(Iteration 601 / 1470) loss: 2.019242
(Iteration 701 / 1470) loss: 2.040140
(Epoch 3 / 6) train acc: 0.340000; val_acc: 0.363000
(Iteration 801 / 1470) loss: 1.963645
(Iteration 901 / 1470) loss: 1.954833
(Epoch 4 / 6) train acc: 0.384000; val_acc: 0.386000
(Iteration 1001 / 1470) loss: 2.034624
(Iteration 1101 / 1470) loss: 1.900691
(Iteration 1201 / 1470) loss: 1.881372
(Epoch 5 / 6) train acc: 0.398000; val_acc: 0.393000
(Iteration 1301 / 1470) loss: 1.822501
(Iteration 1401 / 1470) loss: 1.764563
(Epoch 6 / 6) train acc: 0.429000; val_acc: 0.401000
(Iteration 1 / 978) loss: 2.483739
(Epoch 0 / 6) train acc: 0.141000; val_acc: 0.138000
(Iteration 101 / 978) loss: 2.391559
(Epoch 1 / 6) train acc: 0.273000; val_acc: 0.277000
(Iteration 201 / 978) loss: 2.259250
(Iteration 301 / 978) loss: 2.188280
(Epoch 2 / 6) train acc: 0.299000; val_acc: 0.303000
(Iteration 401 / 978) loss: 2.171115
(Epoch 3 / 6) train acc: 0.320000; val_acc: 0.328000
(Iteration 501 / 978) loss: 2.075260
(Iteration 601 / 978) loss: 2.029646
(Epoch 4 / 6) train acc: 0.351000; val_acc: 0.349000
(Iteration 701 / 978) loss: 2.063060
(Iteration 801 / 978) loss: 2.054358
(Epoch 5 / 6) train acc: 0.394000; val_acc: 0.369000
(Iteration 901 / 978) loss: 1.994019
(Epoch 6 / 6) train acc: 0.403000; val_acc: 0.382000
(Iteration 1 / 1960) loss: 2.454116
(Epoch 0 / 4) train acc: 0.091000; val_acc: 0.099000
(Iteration 101 / 1960) loss: 2.305041
(Iteration 201 / 1960) loss: 2.243288
(Iteration 301 / 1960) loss: 2.135262
(Iteration 401 / 1960) loss: 2.148537
(Epoch 1 / 4) train acc: 0.309000; val_acc: 0.339000
(Iteration 501 / 1960) loss: 2.014377
(Iteration 601 / 1960) loss: 1.986600
(Iteration 701 / 1960) loss: 2.038472
(Iteration 801 / 1960) loss: 2.025236
(Iteration 901 / 1960) loss: 1.729134
(Epoch 2 / 4) train acc: 0.408000; val_acc: 0.390000
(Iteration 1001 / 1960) loss: 1.810217
(Iteration 1101 / 1960) loss: 1.809849
(Iteration 1201 / 1960) loss: 1.841237
```

```
(Iteration 1301 / 1960) loss: 1.809537
(Iteration 1401 / 1960) loss: 1.815509
(Epoch 3 / 4) train acc: 0.400000; val_acc: 0.426000
(Iteration 1501 / 1960) loss: 1.813363
(Iteration 1601 / 1960) loss: 1.678233
(Iteration 1701 / 1960) loss: 1.886531
(Iteration 1801 / 1960) loss: 1.661185
(Iteration 1901 / 1960) loss: 1.724002
(Epoch 4 / 4) train acc: 0.423000; val_acc: 0.448000
(Iteration 1 / 980) loss: 2.441308
(Epoch 0 / 4) train acc: 0.104000; val_acc: 0.123000
(Iteration 101 / 980) loss: 2.325064
(Iteration 201 / 980) loss: 2.181687
(Epoch 1 / 4) train acc: 0.308000; val_acc: 0.306000
(Iteration 301 / 980) loss: 2.115862
(Iteration 401 / 980) loss: 2.048476
(Epoch 2 / 4) train acc: 0.330000; val_acc: 0.353000
(Iteration 501 / 980) loss: 1.959723
(Iteration 601 / 980) loss: 1.922518
(Iteration 701 / 980) loss: 1.832868
(Epoch 3 / 4) train acc: 0.397000; val_acc: 0.385000
(Iteration 801 / 980) loss: 1.914890
(Iteration 901 / 980) loss: 1.947076
(Epoch 4 / 4) train acc: 0.386000; val_acc: 0.400000
(Iteration 1 / 652) loss: 2.440426
(Epoch 0 / 4) train acc: 0.111000; val_acc: 0.119000
(Iteration 101 / 652) loss: 2.329048
(Epoch 1 / 4) train acc: 0.254000; val_acc: 0.293000
(Iteration 201 / 652) loss: 2.164467
(Iteration 301 / 652) loss: 2.081502
(Epoch 2 / 4) train acc: 0.304000; val_acc: 0.320000
(Iteration 401 / 652) loss: 2.006009
(Epoch 3 / 4) train acc: 0.327000; val_acc: 0.341000
(Iteration 501 / 652) loss: 2.079717
(Iteration 601 / 652) loss: 1.957591
(Epoch 4 / 4) train acc: 0.375000; val_acc: 0.369000
(Iteration 1 / 1960) loss: 2.485090
(Epoch 0 / 4) train acc: 0.088000; val_acc: 0.079000
(Iteration 101 / 1960) loss: 2.328048
(Iteration 201 / 1960) loss: 2.278705
(Iteration 301 / 1960) loss: 2.167544
(Iteration 401 / 1960) loss: 2.165839
(Epoch 1 / 4) train acc: 0.310000; val_acc: 0.348000
(Iteration 501 / 1960) loss: 1.928660
(Iteration 601 / 1960) loss: 2.110428
(Iteration 701 / 1960) loss: 1.944180
(Iteration 801 / 1960) loss: 1.953320
(Iteration 901 / 1960) loss: 1.878044
```

```
(Epoch 2 / 4) train acc: 0.408000; val_acc: 0.396000
(Iteration 1001 / 1960) loss: 1.775255
(Iteration 1101 / 1960) loss: 1.869183
(Iteration 1201 / 1960) loss: 2.018306
(Iteration 1301 / 1960) loss: 1.829464
(Iteration 1401 / 1960) loss: 1.883332
(Epoch 3 / 4) train acc: 0.402000; val_acc: 0.426000
(Iteration 1501 / 1960) loss: 1.935965
(Iteration 1601 / 1960) loss: 1.855323
(Iteration 1701 / 1960) loss: 1.869414
(Iteration 1801 / 1960) loss: 1.834183
(Iteration 1901 / 1960) loss: 1.787619
(Epoch 4 / 4) train acc: 0.431000; val_acc: 0.439000
(Iteration 1 / 980) loss: 2.486056
(Epoch 0 / 4) train acc: 0.117000; val_acc: 0.130000
(Iteration 101 / 980) loss: 2.311580
(Iteration 201 / 980) loss: 2.272978
(Epoch 1 / 4) train acc: 0.297000; val_acc: 0.301000
(Iteration 301 / 980) loss: 2.160084
(Iteration 401 / 980) loss: 2.115020
(Epoch 2 / 4) train acc: 0.327000; val_acc: 0.342000
(Iteration 501 / 980) loss: 2.011015
(Iteration 601 / 980) loss: 2.100898
(Iteration 701 / 980) loss: 1.966293
(Epoch 3 / 4) train acc: 0.399000; val_acc: 0.373000
(Iteration 801 / 980) loss: 1.968442
(Iteration 901 / 980) loss: 2.025058
(Epoch 4 / 4) train acc: 0.392000; val_acc: 0.382000
(Iteration 1 / 652) loss: 2.484321
(Epoch 0 / 4) train acc: 0.103000; val_acc: 0.114000
(Iteration 101 / 652) loss: 2.353051
(Epoch 1 / 4) train acc: 0.264000; val_acc: 0.290000
(Iteration 201 / 652) loss: 2.262660
(Iteration 301 / 652) loss: 2.155519
(Epoch 2 / 4) train acc: 0.320000; val_acc: 0.326000
(Iteration 401 / 652) loss: 2.161256
(Epoch 3 / 4) train acc: 0.372000; val_acc: 0.343000
(Iteration 501 / 652) loss: 2.057748
(Iteration 601 / 652) loss: 1.954919
(Epoch 4 / 4) train acc: 0.350000; val_acc: 0.364000
(Iteration 1 / 1960) loss: 2.533901
(Epoch 0 / 4) train acc: 0.100000; val_acc: 0.118000
(Iteration 101 / 1960) loss: 2.401316
(Iteration 201 / 1960) loss: 2.323870
(Iteration 301 / 1960) loss: 2.157719
(Iteration 401 / 1960) loss: 2.136419
(Epoch 1 / 4) train acc: 0.333000; val_acc: 0.343000
(Iteration 501 / 1960) loss: 2.049884
```

```
(Iteration 601 / 1960) loss: 2.246648
(Iteration 701 / 1960) loss: 1.876931
(Iteration 801 / 1960) loss: 2.086051
(Iteration 901 / 1960) loss: 2.104469
(Epoch 2 / 4) train acc: 0.410000; val_acc: 0.393000
(Iteration 1001 / 1960) loss: 1.836167
(Iteration 1101 / 1960) loss: 1.874986
(Iteration 1201 / 1960) loss: 1.775606
(Iteration 1301 / 1960) loss: 1.737154
(Iteration 1401 / 1960) loss: 1.917972
(Epoch 3 / 4) train acc: 0.414000; val_acc: 0.421000
(Iteration 1501 / 1960) loss: 1.984416
(Iteration 1601 / 1960) loss: 1.900833
(Iteration 1701 / 1960) loss: 1.774440
(Iteration 1801 / 1960) loss: 1.830033
(Iteration 1901 / 1960) loss: 1.931057
(Epoch 4 / 4) train acc: 0.445000; val_acc: 0.430000
(Iteration 1 / 980) loss: 2.537888
(Epoch 0 / 4) train acc: 0.126000; val_acc: 0.121000
(Iteration 101 / 980) loss: 2.391795
(Iteration 201 / 980) loss: 2.335238
(Epoch 1 / 4) train acc: 0.295000; val_acc: 0.296000
(Iteration 301 / 980) loss: 2.193771
(Iteration 401 / 980) loss: 2.179341
(Epoch 2 / 4) train acc: 0.314000; val_acc: 0.336000
(Iteration 501 / 980) loss: 2.038463
(Iteration 601 / 980) loss: 2.095191
(Iteration 701 / 980) loss: 2.039363
(Epoch 3 / 4) train acc: 0.364000; val_acc: 0.376000
(Iteration 801 / 980) loss: 1.942650
(Iteration 901 / 980) loss: 1.972401
(Epoch 4 / 4) train acc: 0.381000; val_acc: 0.388000
(Iteration 1 / 652) loss: 2.528392
(Epoch 0 / 4) train acc: 0.097000; val_acc: 0.136000
(Iteration 101 / 652) loss: 2.369533
(Epoch 1 / 4) train acc: 0.252000; val_acc: 0.292000
(Iteration 201 / 652) loss: 2.289078
(Iteration 301 / 652) loss: 2.198953
(Epoch 2 / 4) train acc: 0.321000; val_acc: 0.324000
(Iteration 401 / 652) loss: 2.140513
(Epoch 3 / 4) train acc: 0.339000; val_acc: 0.346000
(Iteration 501 / 652) loss: 2.152863
(Iteration 601 / 652) loss: 2.120479
(Epoch 4 / 4) train acc: 0.346000; val_acc: 0.358000
(Iteration 1 / 1960) loss: 2.579186
(Epoch 0 / 4) train acc: 0.105000; val_acc: 0.080000
(Iteration 101 / 1960) loss: 2.446762
(Iteration 201 / 1960) loss: 2.289352
```

```
(Iteration 301 / 1960) loss: 2.321987
(Iteration 401 / 1960) loss: 2.204697
(Epoch 1 / 4) train acc: 0.363000; val_acc: 0.342000
(Iteration 501 / 1960) loss: 2.115521
(Iteration 601 / 1960) loss: 2.186187
(Iteration 701 / 1960) loss: 2.095841
(Iteration 801 / 1960) loss: 2.051607
(Iteration 901 / 1960) loss: 1.995079
(Epoch 2 / 4) train acc: 0.386000; val_acc: 0.386000
(Iteration 1001 / 1960) loss: 2.084197
(Iteration 1101 / 1960) loss: 1.953538
(Iteration 1201 / 1960) loss: 1.964658
(Iteration 1301 / 1960) loss: 1.908676
(Iteration 1401 / 1960) loss: 1.873376
(Epoch 3 / 4) train acc: 0.424000; val_acc: 0.419000
(Iteration 1501 / 1960) loss: 1.897660
(Iteration 1601 / 1960) loss: 1.990957
(Iteration 1701 / 1960) loss: 1.933465
(Iteration 1801 / 1960) loss: 2.001449
(Iteration 1901 / 1960) loss: 1.780607
(Epoch 4 / 4) train acc: 0.423000; val_acc: 0.430000
(Iteration 1 / 980) loss: 2.580815
(Epoch 0 / 4) train acc: 0.099000; val_acc: 0.089000
(Iteration 101 / 980) loss: 2.428558
(Iteration 201 / 980) loss: 2.316423
(Epoch 1 / 4) train acc: 0.304000; val_acc: 0.304000
(Iteration 301 / 980) loss: 2.244396
(Iteration 401 / 980) loss: 2.138999
(Epoch 2 / 4) train acc: 0.326000; val_acc: 0.348000
(Iteration 501 / 980) loss: 2.140142
(Iteration 601 / 980) loss: 2.172923
(Iteration 701 / 980) loss: 2.051931
(Epoch 3 / 4) train acc: 0.363000; val_acc: 0.377000
(Iteration 801 / 980) loss: 2.006681
(Iteration 901 / 980) loss: 1.957241
(Epoch 4 / 4) train acc: 0.371000; val_acc: 0.395000
(Iteration 1 / 652) loss: 2.581493
(Epoch 0 / 4) train acc: 0.114000; val_acc: 0.106000
(Iteration 101 / 652) loss: 2.428334
(Epoch 1 / 4) train acc: 0.244000; val_acc: 0.286000
(Iteration 201 / 652) loss: 2.296705
(Iteration 301 / 652) loss: 2.225090
(Epoch 2 / 4) train acc: 0.322000; val_acc: 0.314000
(Iteration 401 / 652) loss: 2.235627
(Epoch 3 / 4) train acc: 0.331000; val_acc: 0.339000
(Iteration 501 / 652) loss: 2.192630
(Iteration 601 / 652) loss: 2.074231
(Epoch 4 / 4) train acc: 0.340000; val_acc: 0.363000
```

```
(Iteration 1 / 2450) loss: 2.443773
(Epoch 0 / 5) train acc: 0.073000; val_acc: 0.094000
(Iteration 101 / 2450) loss: 2.299029
(Iteration 201 / 2450) loss: 2.168568
(Iteration 301 / 2450) loss: 2.167773
(Iteration 401 / 2450) loss: 2.064223
(Epoch 1 / 5) train acc: 0.328000; val_acc: 0.344000
(Iteration 501 / 2450) loss: 1.936636
(Iteration 601 / 2450) loss: 2.112930
(Iteration 701 / 2450) loss: 1.978258
(Iteration 801 / 2450) loss: 2.071495
(Iteration 901 / 2450) loss: 1.948586
(Epoch 2 / 5) train acc: 0.411000; val_acc: 0.388000
(Iteration 1001 / 2450) loss: 1.857908
(Iteration 1101 / 2450) loss: 1.788498
(Iteration 1201 / 2450) loss: 1.932100
(Iteration 1301 / 2450) loss: 1.865791
(Iteration 1401 / 2450) loss: 1.853426
(Epoch 3 / 5) train acc: 0.425000; val_acc: 0.416000
(Iteration 1501 / 2450) loss: 1.645963
(Iteration 1601 / 2450) loss: 1.859648
(Iteration 1701 / 2450) loss: 1.711632
(Iteration 1801 / 2450) loss: 1.719462
(Iteration 1901 / 2450) loss: 1.862501
(Epoch 4 / 5) train acc: 0.443000; val_acc: 0.439000
(Iteration 2001 / 2450) loss: 1.620702
(Iteration 2101 / 2450) loss: 1.634766
(Iteration 2201 / 2450) loss: 1.559592
(Iteration 2301 / 2450) loss: 1.355524
(Iteration 2401 / 2450) loss: 1.841314
(Epoch 5 / 5) train acc: 0.466000; val_acc: 0.460000
(Iteration 1 / 1225) loss: 2.442562
(Epoch 0 / 5) train acc: 0.096000; val_acc: 0.084000
(Iteration 101 / 1225) loss: 2.319141
(Iteration 201 / 1225) loss: 2.158181
(Epoch 1 / 5) train acc: 0.280000; val_acc: 0.305000
(Iteration 301 / 1225) loss: 2.089333
(Iteration 401 / 1225) loss: 2.171859
(Epoch 2 / 5) train acc: 0.351000; val_acc: 0.341000
(Iteration 501 / 1225) loss: 1.899862
(Iteration 601 / 1225) loss: 1.959638
(Iteration 701 / 1225) loss: 1.918737
(Epoch 3 / 5) train acc: 0.392000; val_acc: 0.366000
(Iteration 801 / 1225) loss: 1.901342
(Iteration 901 / 1225) loss: 1.932217
(Epoch 4 / 5) train acc: 0.388000; val_acc: 0.387000
(Iteration 1001 / 1225) loss: 1.950619
(Iteration 1101 / 1225) loss: 1.878852
```

```
(Iteration 1201 / 1225) loss: 1.859298
(Epoch 5 / 5) train acc: 0.404000; val_acc: 0.399000
(Iteration 1 / 815) loss: 2.443439
(Epoch 0 / 5) train acc: 0.086000; val_acc: 0.094000
(Iteration 101 / 815) loss: 2.281549
(Epoch 1 / 5) train acc: 0.264000; val_acc: 0.292000
(Iteration 201 / 815) loss: 2.203562
(Iteration 301 / 815) loss: 2.091563
(Epoch 2 / 5) train acc: 0.297000; val_acc: 0.312000
(Iteration 401 / 815) loss: 1.987828
(Epoch 3 / 5) train acc: 0.330000; val_acc: 0.341000
(Iteration 501 / 815) loss: 1.991387
(Iteration 601 / 815) loss: 1.938623
(Epoch 4 / 5) train acc: 0.354000; val_acc: 0.359000
(Iteration 701 / 815) loss: 1.927917
(Iteration 801 / 815) loss: 1.979697
(Epoch 5 / 5) train acc: 0.383000; val_acc: 0.376000
(Iteration 1 / 2450) loss: 2.495668
(Epoch 0 / 5) train acc: 0.100000; val_acc: 0.096000
(Iteration 101 / 2450) loss: 2.400783
(Iteration 201 / 2450) loss: 2.163958
(Iteration 301 / 2450) loss: 2.170425
(Iteration 401 / 2450) loss: 2.048399
(Epoch 1 / 5) train acc: 0.343000; val_acc: 0.342000
(Iteration 501 / 2450) loss: 1.961978
(Iteration 601 / 2450) loss: 2.048626
(Iteration 701 / 2450) loss: 1.913240
(Iteration 801 / 2450) loss: 1.954288
(Iteration 901 / 2450) loss: 2.008229
(Epoch 2 / 5) train acc: 0.392000; val_acc: 0.383000
(Iteration 1001 / 2450) loss: 1.996699
(Iteration 1101 / 2450) loss: 1.988798
(Iteration 1201 / 2450) loss: 1.970870
(Iteration 1301 / 2450) loss: 1.979691
(Iteration 1401 / 2450) loss: 1.861486
(Epoch 3 / 5) train acc: 0.411000; val_acc: 0.420000
(Iteration 1501 / 2450) loss: 1.827895
(Iteration 1601 / 2450) loss: 1.712868
(Iteration 1701 / 2450) loss: 1.731376
(Iteration 1801 / 2450) loss: 1.734315
(Iteration 1901 / 2450) loss: 1.795493
(Epoch 4 / 5) train acc: 0.428000; val_acc: 0.446000
(Iteration 2001 / 2450) loss: 1.767362
(Iteration 2101 / 2450) loss: 1.856515
(Iteration 2201 / 2450) loss: 1.818399
(Iteration 2301 / 2450) loss: 1.598107
(Iteration 2401 / 2450) loss: 1.671781
(Epoch 5 / 5) train acc: 0.443000; val_acc: 0.450000
```

```
(Iteration 1 / 1225) loss: 2.493168
(Epoch 0 / 5) train acc: 0.105000; val_acc: 0.098000
(Iteration 101 / 1225) loss: 2.342500
(Iteration 201 / 1225) loss: 2.225411
(Epoch 1 / 5) train acc: 0.311000; val_acc: 0.313000
(Iteration 301 / 1225) loss: 2.124423
(Iteration 401 / 1225) loss: 2.078431
(Epoch 2 / 5) train acc: 0.338000; val_acc: 0.345000
(Iteration 501 / 1225) loss: 2.041322
(Iteration 601 / 1225) loss: 1.963422
(Iteration 701 / 1225) loss: 1.953043
(Epoch 3 / 5) train acc: 0.361000; val_acc: 0.376000
(Iteration 801 / 1225) loss: 1.954553
(Iteration 901 / 1225) loss: 1.942476
(Epoch 4 / 5) train acc: 0.395000; val_acc: 0.392000
(Iteration 1001 / 1225) loss: 1.902198
(Iteration 1101 / 1225) loss: 1.947481
(Iteration 1201 / 1225) loss: 1.869607
(Epoch 5 / 5) train acc: 0.415000; val_acc: 0.414000
(Iteration 1 / 815) loss: 2.483857
(Epoch 0 / 5) train acc: 0.130000; val_acc: 0.136000
(Iteration 101 / 815) loss: 2.366373
(Epoch 1 / 5) train acc: 0.262000; val_acc: 0.281000
(Iteration 201 / 815) loss: 2.198449
(Iteration 301 / 815) loss: 2.154394
(Epoch 2 / 5) train acc: 0.313000; val_acc: 0.315000
(Iteration 401 / 815) loss: 2.099643
(Epoch 3 / 5) train acc: 0.334000; val_acc: 0.327000
(Iteration 501 / 815) loss: 2.068800
(Iteration 601 / 815) loss: 1.908750
(Epoch 4 / 5) train acc: 0.382000; val_acc: 0.363000
(Iteration 701 / 815) loss: 2.031011
(Iteration 801 / 815) loss: 1.965732
(Epoch 5 / 5) train acc: 0.371000; val_acc: 0.385000
(Iteration 1 / 2450) loss: 2.532137
(Epoch 0 / 5) train acc: 0.079000; val_acc: 0.100000
(Iteration 101 / 2450) loss: 2.416548
(Iteration 201 / 2450) loss: 2.226517
(Iteration 301 / 2450) loss: 2.177370
(Iteration 401 / 2450) loss: 2.139111
(Epoch 1 / 5) train acc: 0.352000; val_acc: 0.345000
(Iteration 501 / 2450) loss: 2.171956
(Iteration 601 / 2450) loss: 2.065615
(Iteration 701 / 2450) loss: 2.226926
(Iteration 801 / 2450) loss: 1.996923
(Iteration 901 / 2450) loss: 1.940391
(Epoch 2 / 5) train acc: 0.393000; val_acc: 0.383000
(Iteration 1001 / 2450) loss: 1.980103
```

```
(Iteration 1101 / 2450) loss: 1.945167
(Iteration 1201 / 2450) loss: 1.809234
(Iteration 1301 / 2450) loss: 1.858808
(Iteration 1401 / 2450) loss: 2.146207
(Epoch 3 / 5) train acc: 0.420000; val_acc: 0.418000
(Iteration 1501 / 2450) loss: 1.945274
(Iteration 1601 / 2450) loss: 1.812496
(Iteration 1701 / 2450) loss: 1.778492
(Iteration 1801 / 2450) loss: 1.961188
(Iteration 1901 / 2450) loss: 1.844633
(Epoch 4 / 5) train acc: 0.452000; val_acc: 0.442000
(Iteration 2001 / 2450) loss: 1.921316
(Iteration 2101 / 2450) loss: 1.854688
(Iteration 2201 / 2450) loss: 1.828423
(Iteration 2301 / 2450) loss: 1.785738
(Iteration 2401 / 2450) loss: 1.770947
(Epoch 5 / 5) train acc: 0.442000; val_acc: 0.452000
(Iteration 1 / 1225) loss: 2.529650
(Epoch 0 / 5) train acc: 0.103000; val_acc: 0.135000
(Iteration 101 / 1225) loss: 2.390803
(Iteration 201 / 1225) loss: 2.356255
(Epoch 1 / 5) train acc: 0.300000; val_acc: 0.304000
(Iteration 301 / 1225) loss: 2.260589
(Iteration 401 / 1225) loss: 2.198031
(Epoch 2 / 5) train acc: 0.345000; val_acc: 0.335000
(Iteration 501 / 1225) loss: 2.047060
(Iteration 601 / 1225) loss: 2.108001
(Iteration 701 / 1225) loss: 1.998871
(Epoch 3 / 5) train acc: 0.350000; val_acc: 0.368000
(Iteration 801 / 1225) loss: 2.044496
(Iteration 901 / 1225) loss: 2.003761
(Epoch 4 / 5) train acc: 0.393000; val_acc: 0.384000
(Iteration 1001 / 1225) loss: 1.892740
(Iteration 1101 / 1225) loss: 1.966783
(Iteration 1201 / 1225) loss: 1.896369
(Epoch 5 / 5) train acc: 0.402000; val_acc: 0.394000
(Iteration 1 / 815) loss: 2.534576
(Epoch 0 / 5) train acc: 0.098000; val_acc: 0.093000
(Iteration 101 / 815) loss: 2.383221
(Epoch 1 / 5) train acc: 0.235000; val_acc: 0.288000
(Iteration 201 / 815) loss: 2.281534
(Iteration 301 / 815) loss: 2.256019
(Epoch 2 / 5) train acc: 0.334000; val_acc: 0.329000
(Iteration 401 / 815) loss: 2.154055
(Epoch 3 / 5) train acc: 0.349000; val_acc: 0.350000
(Iteration 501 / 815) loss: 2.163742
(Iteration 601 / 815) loss: 2.096310
(Epoch 4 / 5) train acc: 0.360000; val_acc: 0.369000
```

```
(Iteration 701 / 815) loss: 2.041314
(Iteration 801 / 815) loss: 1.933947
(Epoch 5 / 5) train acc: 0.385000; val_acc: 0.377000
(Iteration 1 / 2450) loss: 2.580960
(Epoch 0 / 5) train acc: 0.101000; val_acc: 0.097000
(Iteration 101 / 2450) loss: 2.420447
(Iteration 201 / 2450) loss: 2.372846
(Iteration 301 / 2450) loss: 2.209834
(Iteration 401 / 2450) loss: 2.229410
(Epoch 1 / 5) train acc: 0.347000; val_acc: 0.334000
(Iteration 501 / 2450) loss: 2.085374
(Iteration 601 / 2450) loss: 1.952153
(Iteration 701 / 2450) loss: 1.918031
(Iteration 801 / 2450) loss: 2.020160
(Iteration 901 / 2450) loss: 1.931854
(Epoch 2 / 5) train acc: 0.386000; val_acc: 0.399000
(Iteration 1001 / 2450) loss: 1.891793
(Iteration 1101 / 2450) loss: 2.075963
(Iteration 1201 / 2450) loss: 2.004080
(Iteration 1301 / 2450) loss: 1.822485
(Iteration 1401 / 2450) loss: 1.942278
(Epoch 3 / 5) train acc: 0.429000; val_acc: 0.418000
(Iteration 1501 / 2450) loss: 2.045888
(Iteration 1601 / 2450) loss: 1.981855
(Iteration 1701 / 2450) loss: 2.044040
(Iteration 1801 / 2450) loss: 1.865390
(Iteration 1901 / 2450) loss: 1.816596
(Epoch 4 / 5) train acc: 0.429000; val_acc: 0.452000
(Iteration 2001 / 2450) loss: 1.698543
(Iteration 2101 / 2450) loss: 1.765890
(Iteration 2201 / 2450) loss: 1.820805
(Iteration 2301 / 2450) loss: 1.905248
(Iteration 2401 / 2450) loss: 2.042233
(Epoch 5 / 5) train acc: 0.463000; val_acc: 0.449000
(Iteration 1 / 1225) loss: 2.581265
(Epoch 0 / 5) train acc: 0.108000; val_acc: 0.113000
(Iteration 101 / 1225) loss: 2.420627
(Iteration 201 / 1225) loss: 2.336231
(Epoch 1 / 5) train acc: 0.255000; val_acc: 0.304000
(Iteration 301 / 1225) loss: 2.219192
(Iteration 401 / 1225) loss: 2.263289
(Epoch 2 / 5) train acc: 0.331000; val_acc: 0.338000
(Iteration 501 / 1225) loss: 2.121514
(Iteration 601 / 1225) loss: 2.084963
(Iteration 701 / 1225) loss: 2.027703
(Epoch 3 / 5) train acc: 0.368000; val_acc: 0.376000
(Iteration 801 / 1225) loss: 2.084090
(Iteration 901 / 1225) loss: 2.105452
```

```
(Epoch 4 / 5) train acc: 0.386000; val_acc: 0.398000
(Iteration 1001 / 1225) loss: 1.971036
(Iteration 1101 / 1225) loss: 1.931570
(Iteration 1201 / 1225) loss: 1.981979
(Epoch 5 / 5) train acc: 0.424000; val_acc: 0.401000
(Iteration 1 / 815) loss: 2.581575
(Epoch 0 / 5) train acc: 0.080000; val_acc: 0.095000
(Iteration 101 / 815) loss: 2.436831
(Epoch 1 / 5) train acc: 0.266000; val_acc: 0.299000
(Iteration 201 / 815) loss: 2.343850
(Iteration 301 / 815) loss: 2.218815
(Epoch 2 / 5) train acc: 0.301000; val_acc: 0.326000
(Iteration 401 / 815) loss: 2.210110
(Epoch 3 / 5) train acc: 0.313000; val_acc: 0.342000
(Iteration 501 / 815) loss: 2.160202
(Iteration 601 / 815) loss: 2.032099
(Epoch 4 / 5) train acc: 0.368000; val_acc: 0.367000
(Iteration 701 / 815) loss: 2.011467
(Iteration 801 / 815) loss: 2.114812
(Epoch 5 / 5) train acc: 0.366000; val_acc: 0.386000
(Iteration 1 / 2940) loss: 2.441493
(Epoch 0 / 6) train acc: 0.089000; val_acc: 0.082000
(Iteration 101 / 2940) loss: 2.324304
(Iteration 201 / 2940) loss: 2.127419
(Iteration 301 / 2940) loss: 2.133023
(Iteration 401 / 2940) loss: 2.148690
(Epoch 1 / 6) train acc: 0.320000; val_acc: 0.345000
(Iteration 501 / 2940) loss: 1.987740
(Iteration 601 / 2940) loss: 1.909199
(Iteration 701 / 2940) loss: 1.935418
(Iteration 801 / 2940) loss: 2.062869
(Iteration 901 / 2940) loss: 1.880897
(Epoch 2 / 6) train acc: 0.388000; val_acc: 0.391000
(Iteration 1001 / 2940) loss: 1.904916
(Iteration 1101 / 2940) loss: 1.778226
(Iteration 1201 / 2940) loss: 1.703566
(Iteration 1301 / 2940) loss: 1.963180
(Iteration 1401 / 2940) loss: 1.822867
(Epoch 3 / 6) train acc: 0.417000; val_acc: 0.418000
(Iteration 1501 / 2940) loss: 1.757183
(Iteration 1601 / 2940) loss: 1.769165
(Iteration 1701 / 2940) loss: 1.764545
(Iteration 1801 / 2940) loss: 1.770595
(Iteration 1901 / 2940) loss: 1.744908
(Epoch 4 / 6) train acc: 0.418000; val_acc: 0.430000
(Iteration 2001 / 2940) loss: 1.700567
(Iteration 2101 / 2940) loss: 1.720707
(Iteration 2201 / 2940) loss: 1.765825
```

```
(Iteration 2301 / 2940) loss: 1.698556
(Iteration 2401 / 2940) loss: 1.590259
(Epoch 5 / 6) train acc: 0.432000; val_acc: 0.455000
(Iteration 2501 / 2940) loss: 1.855911
(Iteration 2601 / 2940) loss: 1.738511
(Iteration 2701 / 2940) loss: 1.680553
(Iteration 2801 / 2940) loss: 1.598512
(Iteration 2901 / 2940) loss: 1.744634
(Epoch 6 / 6) train acc: 0.491000; val_acc: 0.458000
(Iteration 1 / 1470) loss: 2.445057
(Epoch 0 / 6) train acc: 0.115000; val_acc: 0.119000
(Iteration 101 / 1470) loss: 2.292423
(Iteration 201 / 1470) loss: 2.167510
(Epoch 1 / 6) train acc: 0.262000; val_acc: 0.301000
(Iteration 301 / 1470) loss: 2.103008
(Iteration 401 / 1470) loss: 2.100337
(Epoch 2 / 6) train acc: 0.319000; val_acc: 0.337000
(Iteration 501 / 1470) loss: 1.981930
(Iteration 601 / 1470) loss: 2.007980
(Iteration 701 / 1470) loss: 1.897695
(Epoch 3 / 6) train acc: 0.380000; val_acc: 0.380000
(Iteration 801 / 1470) loss: 2.054148
(Iteration 901 / 1470) loss: 1.876522
(Epoch 4 / 6) train acc: 0.388000; val_acc: 0.389000
(Iteration 1001 / 1470) loss: 1.899088
(Iteration 1101 / 1470) loss: 1.886730
(Iteration 1201 / 1470) loss: 1.806114
(Epoch 5 / 6) train acc: 0.410000; val_acc: 0.403000
(Iteration 1301 / 1470) loss: 1.903311
(Iteration 1401 / 1470) loss: 1.747794
(Epoch 6 / 6) train acc: 0.419000; val_acc: 0.429000
(Iteration 1 / 978) loss: 2.441519
(Epoch 0 / 6) train acc: 0.098000; val_acc: 0.100000
(Iteration 101 / 978) loss: 2.313660
(Epoch 1 / 6) train acc: 0.274000; val_acc: 0.285000
(Iteration 201 / 978) loss: 2.207172
(Iteration 301 / 978) loss: 2.135085
(Epoch 2 / 6) train acc: 0.330000; val_acc: 0.319000
(Iteration 401 / 978) loss: 2.065399
(Epoch 3 / 6) train acc: 0.335000; val_acc: 0.336000
(Iteration 501 / 978) loss: 2.017478
(Iteration 601 / 978) loss: 2.051684
(Epoch 4 / 6) train acc: 0.357000; val_acc: 0.365000
(Iteration 701 / 978) loss: 1.862432
(Iteration 801 / 978) loss: 1.955985
(Epoch 5 / 6) train acc: 0.383000; val_acc: 0.382000
(Iteration 901 / 978) loss: 1.845738
(Epoch 6 / 6) train acc: 0.361000; val_acc: 0.394000
```

```
(Iteration 1 / 2940) loss: 2.484509
(Epoch 0 / 6) train acc: 0.114000; val_acc: 0.118000
(Iteration 101 / 2940) loss: 2.362523
(Iteration 201 / 2940) loss: 2.304916
(Iteration 301 / 2940) loss: 2.148371
(Iteration 401 / 2940) loss: 2.129030
(Epoch 1 / 6) train acc: 0.340000; val_acc: 0.341000
(Iteration 501 / 2940) loss: 1.970671
(Iteration 601 / 2940) loss: 2.011186
(Iteration 701 / 2940) loss: 1.957754
(Iteration 801 / 2940) loss: 1.986204
(Iteration 901 / 2940) loss: 1.913199
(Epoch 2 / 6) train acc: 0.399000; val_acc: 0.386000
(Iteration 1001 / 2940) loss: 1.990815
(Iteration 1101 / 2940) loss: 2.089271
(Iteration 1201 / 2940) loss: 1.968640
(Iteration 1301 / 2940) loss: 1.841899
(Iteration 1401 / 2940) loss: 1.795296
(Epoch 3 / 6) train acc: 0.433000; val_acc: 0.414000
(Iteration 1501 / 2940) loss: 1.959468
(Iteration 1601 / 2940) loss: 1.742363
(Iteration 1701 / 2940) loss: 1.748605
(Iteration 1801 / 2940) loss: 1.937646
(Iteration 1901 / 2940) loss: 1.855438
(Epoch 4 / 6) train acc: 0.452000; val_acc: 0.446000
(Iteration 2001 / 2940) loss: 1.893325
(Iteration 2101 / 2940) loss: 1.674625
(Iteration 2201 / 2940) loss: 2.040573
(Iteration 2301 / 2940) loss: 1.786504
(Iteration 2401 / 2940) loss: 1.807196
(Epoch 5 / 6) train acc: 0.488000; val_acc: 0.450000
(Iteration 2501 / 2940) loss: 1.714938
(Iteration 2601 / 2940) loss: 1.651877
(Iteration 2701 / 2940) loss: 1.836652
(Iteration 2801 / 2940) loss: 1.844435
(Iteration 2901 / 2940) loss: 1.643023
(Epoch 6 / 6) train acc: 0.473000; val_acc: 0.451000
(Iteration 1 / 1470) loss: 2.487712
(Epoch 0 / 6) train acc: 0.104000; val_acc: 0.109000
(Iteration 101 / 1470) loss: 2.365178
(Iteration 201 / 1470) loss: 2.208881
(Epoch 1 / 6) train acc: 0.279000; val_acc: 0.311000
(Iteration 301 / 1470) loss: 2.090273
(Iteration 401 / 1470) loss: 2.168254
(Epoch 2 / 6) train acc: 0.335000; val_acc: 0.347000
(Iteration 501 / 1470) loss: 1.992300
(Iteration 601 / 1470) loss: 2.029803
(Iteration 701 / 1470) loss: 2.016693
```

```
(Epoch 3 / 6) train acc: 0.342000; val_acc: 0.371000
(Iteration 801 / 1470) loss: 2.037419
(Iteration 901 / 1470) loss: 1.892423
(Epoch 4 / 6) train acc: 0.387000; val_acc: 0.392000
(Iteration 1001 / 1470) loss: 1.988653
(Iteration 1101 / 1470) loss: 1.844476
(Iteration 1201 / 1470) loss: 1.909789
(Epoch 5 / 6) train acc: 0.392000; val_acc: 0.401000
(Iteration 1301 / 1470) loss: 1.798885
(Iteration 1401 / 1470) loss: 1.920462
(Epoch 6 / 6) train acc: 0.412000; val_acc: 0.422000
(Iteration 1 / 978) loss: 2.487205
(Epoch 0 / 6) train acc: 0.117000; val_acc: 0.126000
(Iteration 101 / 978) loss: 2.318199
(Epoch 1 / 6) train acc: 0.270000; val_acc: 0.288000
(Iteration 201 / 978) loss: 2.253817
(Iteration 301 / 978) loss: 2.101102
(Epoch 2 / 6) train acc: 0.307000; val_acc: 0.315000
(Iteration 401 / 978) loss: 2.090159
(Epoch 3 / 6) train acc: 0.334000; val_acc: 0.336000
(Iteration 501 / 978) loss: 2.115408
(Iteration 601 / 978) loss: 1.992976
(Epoch 4 / 6) train acc: 0.363000; val_acc: 0.359000
(Iteration 701 / 978) loss: 2.083981
(Iteration 801 / 978) loss: 1.993317
(Epoch 5 / 6) train acc: 0.374000; val_acc: 0.373000
(Iteration 901 / 978) loss: 1.965886
(Epoch 6 / 6) train acc: 0.377000; val_acc: 0.386000
(Iteration 1 / 2940) loss: 2.530338
(Epoch 0 / 6) train acc: 0.116000; val_acc: 0.112000
(Iteration 101 / 2940) loss: 2.443428
(Iteration 201 / 2940) loss: 2.302060
(Iteration 301 / 2940) loss: 2.280774
(Iteration 401 / 2940) loss: 2.017572
(Epoch 1 / 6) train acc: 0.366000; val_acc: 0.353000
(Iteration 501 / 2940) loss: 2.131554
(Iteration 601 / 2940) loss: 2.041736
(Iteration 701 / 2940) loss: 2.096603
(Iteration 801 / 2940) loss: 1.998212
(Iteration 901 / 2940) loss: 1.783561
(Epoch 2 / 6) train acc: 0.401000; val_acc: 0.400000
(Iteration 1001 / 2940) loss: 1.941963
(Iteration 1101 / 2940) loss: 1.929360
(Iteration 1201 / 2940) loss: 1.815697
(Iteration 1301 / 2940) loss: 1.832918
(Iteration 1401 / 2940) loss: 1.804996
(Epoch 3 / 6) train acc: 0.410000; val_acc: 0.428000
(Iteration 1501 / 2940) loss: 1.842263
```

```
(Iteration 1601 / 2940) loss: 1.692541
(Iteration 1701 / 2940) loss: 1.885592
(Iteration 1801 / 2940) loss: 1.705028
(Iteration 1901 / 2940) loss: 1.811747
(Epoch 4 / 6) train acc: 0.439000; val_acc: 0.442000
(Iteration 2001 / 2940) loss: 1.885906
(Iteration 2101 / 2940) loss: 1.600432
(Iteration 2201 / 2940) loss: 1.928067
(Iteration 2301 / 2940) loss: 1.842707
(Iteration 2401 / 2940) loss: 1.770090
(Epoch 5 / 6) train acc: 0.455000; val_acc: 0.460000
(Iteration 2501 / 2940) loss: 1.862582
(Iteration 2601 / 2940) loss: 1.752952
(Iteration 2701 / 2940) loss: 2.027475
(Iteration 2801 / 2940) loss: 1.772843
(Iteration 2901 / 2940) loss: 1.839308
(Epoch 6 / 6) train acc: 0.486000; val_acc: 0.468000
(Iteration 1 / 1470) loss: 2.539755
(Epoch 0 / 6) train acc: 0.110000; val_acc: 0.108000
(Iteration 101 / 1470) loss: 2.378957
(Iteration 201 / 1470) loss: 2.299750
(Epoch 1 / 6) train acc: 0.334000; val_acc: 0.320000
(Iteration 301 / 1470) loss: 2.150726
(Iteration 401 / 1470) loss: 2.089958
(Epoch 2 / 6) train acc: 0.350000; val_acc: 0.351000
(Iteration 501 / 1470) loss: 2.050357
(Iteration 601 / 1470) loss: 2.004484
(Iteration 701 / 1470) loss: 2.069726
(Epoch 3 / 6) train acc: 0.372000; val_acc: 0.381000
(Iteration 801 / 1470) loss: 1.952575
(Iteration 901 / 1470) loss: 1.976878
(Epoch 4 / 6) train acc: 0.398000; val_acc: 0.385000
(Iteration 1001 / 1470) loss: 1.937548
(Iteration 1101 / 1470) loss: 1.968524
(Iteration 1201 / 1470) loss: 1.975010
(Epoch 5 / 6) train acc: 0.448000; val_acc: 0.401000
(Iteration 1301 / 1470) loss: 1.863565
(Iteration 1401 / 1470) loss: 1.907946
(Epoch 6 / 6) train acc: 0.422000; val_acc: 0.416000
(Iteration 1 / 978) loss: 2.534772
(Epoch 0 / 6) train acc: 0.114000; val_acc: 0.108000
(Iteration 101 / 978) loss: 2.369671
(Epoch 1 / 6) train acc: 0.263000; val_acc: 0.286000
(Iteration 201 / 978) loss: 2.304269
(Iteration 301 / 978) loss: 2.204693
(Epoch 2 / 6) train acc: 0.305000; val_acc: 0.306000
(Iteration 401 / 978) loss: 2.171596
(Epoch 3 / 6) train acc: 0.361000; val_acc: 0.330000
```

```
(Iteration 501 / 978) loss: 2.051045
(Iteration 601 / 978) loss: 2.076332
(Epoch 4 / 6) train acc: 0.376000; val_acc: 0.352000
(Iteration 701 / 978) loss: 2.047781
(Iteration 801 / 978) loss: 2.011845
(Epoch 5 / 6) train acc: 0.358000; val_acc: 0.374000
(Iteration 901 / 978) loss: 1.945381
(Epoch 6 / 6) train acc: 0.409000; val_acc: 0.384000
(Iteration 1 / 2940) loss: 2.583244
(Epoch 0 / 6) train acc: 0.131000; val_acc: 0.115000
(Iteration 101 / 2940) loss: 2.419673
(Iteration 201 / 2940) loss: 2.373688
(Iteration 301 / 2940) loss: 2.243525
(Iteration 401 / 2940) loss: 2.151651
(Epoch 1 / 6) train acc: 0.342000; val_acc: 0.335000
(Iteration 501 / 2940) loss: 2.063756
(Iteration 601 / 2940) loss: 2.042330
(Iteration 701 / 2940) loss: 1.954390
(Iteration 801 / 2940) loss: 2.009360
(Iteration 901 / 2940) loss: 1.958587
(Epoch 2 / 6) train acc: 0.408000; val_acc: 0.389000
(Iteration 1001 / 2940) loss: 2.028550
(Iteration 1101 / 2940) loss: 2.096986
(Iteration 1201 / 2940) loss: 1.865157
(Iteration 1301 / 2940) loss: 1.911459
(Iteration 1401 / 2940) loss: 1.829405
(Epoch 3 / 6) train acc: 0.395000; val_acc: 0.423000
(Iteration 1501 / 2940) loss: 1.954539
(Iteration 1601 / 2940) loss: 1.738689
(Iteration 1701 / 2940) loss: 1.885682
(Iteration 1801 / 2940) loss: 1.873238
(Iteration 1901 / 2940) loss: 1.933580
(Epoch 4 / 6) train acc: 0.457000; val_acc: 0.447000
(Iteration 2001 / 2940) loss: 1.667659
(Iteration 2101 / 2940) loss: 1.949022
(Iteration 2201 / 2940) loss: 1.778308
(Iteration 2301 / 2940) loss: 1.862346
(Iteration 2401 / 2940) loss: 1.895816
(Epoch 5 / 6) train acc: 0.443000; val_acc: 0.461000
(Iteration 2501 / 2940) loss: 1.887220
(Iteration 2601 / 2940) loss: 1.753012
(Iteration 2701 / 2940) loss: 1.681981
(Iteration 2801 / 2940) loss: 1.912191
(Iteration 2901 / 2940) loss: 1.732678
(Epoch 6 / 6) train acc: 0.487000; val_acc: 0.473000
(Iteration 1 / 1470) loss: 2.574993
(Epoch 0 / 6) train acc: 0.116000; val_acc: 0.113000
(Iteration 101 / 1470) loss: 2.420893
```

```
(Iteration 201 / 1470) loss: 2.312932
(Epoch 1 / 6) train acc: 0.320000; val_acc: 0.306000
(Iteration 301 / 1470) loss: 2.343388
(Iteration 401 / 1470) loss: 2.187478
(Epoch 2 / 6) train acc: 0.334000; val_acc: 0.337000
(Iteration 501 / 1470) loss: 2.151956
(Iteration 601 / 1470) loss: 2.050323
(Iteration 701 / 1470) loss: 2.255854
(Epoch 3 / 6) train acc: 0.365000; val_acc: 0.367000
(Iteration 801 / 1470) loss: 1.955970
(Iteration 901 / 1470) loss: 2.045659
(Epoch 4 / 6) train acc: 0.384000; val_acc: 0.399000
(Iteration 1001 / 1470) loss: 1.976308
(Iteration 1101 / 1470) loss: 1.946492
(Iteration 1201 / 1470) loss: 1.797447
(Epoch 5 / 6) train acc: 0.397000; val_acc: 0.390000
(Iteration 1301 / 1470) loss: 2.118066
(Iteration 1401 / 1470) loss: 2.029960
(Epoch 6 / 6) train acc: 0.404000; val_acc: 0.417000
(Iteration 1 / 978) loss: 2.583246
(Epoch 0 / 6) train acc: 0.130000; val_acc: 0.119000
(Iteration 101 / 978) loss: 2.420848
(Epoch 1 / 6) train acc: 0.267000; val_acc: 0.292000
(Iteration 201 / 978) loss: 2.310696
(Iteration 301 / 978) loss: 2.249860
(Epoch 2 / 6) train acc: 0.325000; val_acc: 0.308000
(Iteration 401 / 978) loss: 2.158943
(Epoch 3 / 6) train acc: 0.345000; val_acc: 0.324000
(Iteration 501 / 978) loss: 2.170879
(Iteration 601 / 978) loss: 2.030965
(Epoch 4 / 6) train acc: 0.384000; val_acc: 0.363000
(Iteration 701 / 978) loss: 2.011998
(Iteration 801 / 978) loss: 2.121224
(Epoch 5 / 6) train acc: 0.356000; val_acc: 0.382000
(Iteration 901 / 978) loss: 2.004404
(Epoch 6 / 6) train acc: 0.383000; val_acc: 0.379000
```

12 Test your model!

Run your best model on the validation and test sets. You should achieve above 48% accuracy on the validation set and the test set.

```
[15]: y_val_pred = np.argmax(best_model.loss(data['X_val']), axis=1)
      print('Validation set accuracy: ', (y_val_pred == data['y_val']).mean())
```

```
Validation set accuracy:  0.525
```

```
[16]: y_test_pred = np.argmax(best_model.loss(data['X_test']), axis=1)
      print('Test set accuracy: ', (y_test_pred == data['y_test']).mean())
```

Test set accuracy: 0.512

```
[17]: # Save best model
      best_model.save("best_two_layer_net.npy")
```

best_two_layer_net.npy saved.

12.1   Inline Question 2:

Now that you have trained a Neural Network classifier, you may find that your testing accuracy is much lower than the training accuracy. In what ways can we decrease this gap? Select all that apply.

1. Train on a larger dataset.
2. Add more hidden units.
3. Increase the regularization strength.
4. None of the above.

Your Answer : 1 & 3

Your Explanation : 1. larger dataset requires the model to generalize patterns more, meaning it is less likely to "memorize" noise/non general paterns. 2. makes the model more complex, making it more likely to overfit! 3. penailizes the model for being to "complex" - meaning it penalizes have very complex deciision boundries, which are corolated to over fitting. 4. clearly wrong.

[17]:

features

December 18, 2025

```python
[1]: # This mounts your Google Drive to the Colab VM.
     from google.colab import drive
     drive.mount('/content/drive')

     # TODO: Enter the foldername in your Drive where you have saved the unzipped
     # assignment folder, e.g. 'icv83551/assignments/assignment1/'
     FOLDERNAME = 'icv83551/assignments/assignment1/'
     assert FOLDERNAME is not None, "[!] Enter the foldername."

     # Now that we've mounted your Drive, this ensures that
     # the Python interpreter of the Colab VM can load
     # python files from within it.
     import sys
     sys.path.append('/content/drive/My Drive/{}'.format(FOLDERNAME))

     # This downloads the CIFAR-10 dataset to your Drive
     # if it doesn't already exist.
     %cd /content/drive/My\ Drive/$FOLDERNAME/icv83551/datasets/
     !bash get_datasets.sh
     %cd /content/drive/My\ Drive/$FOLDERNAME
```

```
Mounted at /content/drive
/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/datasets
/content/drive/My Drive/icv83551/assignments/assignment1
```

1 Image features exercise

*Complete and hand in this completed worksheet (including its outputs and any supporting code outside of the worksheet) with your assignment submission.

We have seen that we can achieve reasonable performance on an image classification task by training a linear classifier on the pixels of the input image. In this exercise we will show that we can improve our classification performance by training linear classifiers not on raw pixels but on features that are computed from the raw pixels.

All of your work for this exercise will be done in this notebook.

```
[2]: import random
     import numpy as np
     from icv83551.data_utils import load_CIFAR10
     import matplotlib.pyplot as plt


     %matplotlib inline
     plt.rcParams['figure.figsize'] = (10.0, 8.0) # set default size of plots
     plt.rcParams['image.interpolation'] = 'nearest'
     plt.rcParams['image.cmap'] = 'gray'

     #%load_ext autoreload
     #%autoreload 2
```

1.1 Load data

Similar to previous exercises, we will load CIFAR-10 data from disk.

```
[3]: from icv83551.features import color_histogram_hsv, hog_feature


     def get_CIFAR10_data(num_training=49000, num_validation=1000, num_test=1000):
         # Load the raw CIFAR-10 data
         cifar10_dir = 'icv83551/datasets/cifar-10-batches-py'

         # Cleaning up variables to prevent loading data multiple times (which may
      ↪cause memory issue)
         try:
             del X_train, y_train
             del X_test, y_test
             print('Clear previously loaded data.')
         except:
             pass

         X_train, y_train, X_test, y_test = load_CIFAR10(cifar10_dir)

         # Subsample the data
         mask = list(range(num_training, num_training + num_validation))
         X_val = X_train[mask]
         y_val = y_train[mask]
         mask = list(range(num_training))
         X_train = X_train[mask]
         y_train = y_train[mask]
         mask = list(range(num_test))
         X_test = X_test[mask]
         y_test = y_test[mask]

         return X_train, y_train, X_val, y_val, X_test, y_test
```

```
X_train, y_train, X_val, y_val, X_test, y_test = get_CIFAR10_data()
```

1.2 Extract Features

For each image we will compute a Histogram of Oriented Gradients (HOG) as well as a color histogram using the hue channel in HSV color space. We form our final feature vector for each image by concatenating the HOG and color histogram feature vectors.

Roughly speaking, HOG should capture the texture of the image while ignoring color information, and the color histogram represents the color of the input image while ignoring texture. As a result, we expect that using both together ought to work better than using either alone. Verifying this assumption would be a good thing to try for your own interest.

The `hog_feature` and `color_histogram_hsv` functions both operate on a single image and return a feature vector for that image. The extract_features function takes a set of images and a list of feature functions and evaluates each feature function on each image, storing the results in a matrix where each column is the concatenation of all feature vectors for a single image.

```python
[4]: from icv83551.features import *


     # num_color_bins = 10 # Number of bins in the color histogram
     num_color_bins = 25 # Number of bins in the color histogram
     feature_fns = [hog_feature, lambda img: color_histogram_hsv(img,␣
      ↪nbin=num_color_bins)]
     X_train_feats = extract_features(X_train, feature_fns, verbose=True)
     X_val_feats = extract_features(X_val, feature_fns)
     X_test_feats = extract_features(X_test, feature_fns)

     # Preprocessing: Subtract the mean feature
     mean_feat = np.mean(X_train_feats, axis=0, keepdims=True)
     X_train_feats -= mean_feat
     X_val_feats -= mean_feat
     X_test_feats -= mean_feat

     # Preprocessing: Divide by standard deviation. This ensures that each feature
     # has roughly the same scale.
     std_feat = np.std(X_train_feats, axis=0, keepdims=True)
     X_train_feats /= std_feat
     X_val_feats /= std_feat
     X_test_feats /= std_feat

     # Preprocessing: Add a bias dimension
     X_train_feats = np.hstack([X_train_feats, np.ones((X_train_feats.shape[0], 1))])
     X_val_feats = np.hstack([X_val_feats, np.ones((X_val_feats.shape[0], 1))])
     X_test_feats = np.hstack([X_test_feats, np.ones((X_test_feats.shape[0], 1))])
```

```
Done extracting features for 1000 / 49000 images
```

3

```
Done extracting features for 2000 / 49000 images
Done extracting features for 3000 / 49000 images
Done extracting features for 4000 / 49000 images
Done extracting features for 5000 / 49000 images
Done extracting features for 6000 / 49000 images
Done extracting features for 7000 / 49000 images
Done extracting features for 8000 / 49000 images
Done extracting features for 9000 / 49000 images
Done extracting features for 10000 / 49000 images
Done extracting features for 11000 / 49000 images
Done extracting features for 12000 / 49000 images
Done extracting features for 13000 / 49000 images
Done extracting features for 14000 / 49000 images
Done extracting features for 15000 / 49000 images
Done extracting features for 16000 / 49000 images
Done extracting features for 17000 / 49000 images
Done extracting features for 18000 / 49000 images
Done extracting features for 19000 / 49000 images
Done extracting features for 20000 / 49000 images
Done extracting features for 21000 / 49000 images
Done extracting features for 22000 / 49000 images
Done extracting features for 23000 / 49000 images
Done extracting features for 24000 / 49000 images
Done extracting features for 25000 / 49000 images
Done extracting features for 26000 / 49000 images
Done extracting features for 27000 / 49000 images
Done extracting features for 28000 / 49000 images
Done extracting features for 29000 / 49000 images
Done extracting features for 30000 / 49000 images
Done extracting features for 31000 / 49000 images
Done extracting features for 32000 / 49000 images
Done extracting features for 33000 / 49000 images
Done extracting features for 34000 / 49000 images
Done extracting features for 35000 / 49000 images
Done extracting features for 36000 / 49000 images
Done extracting features for 37000 / 49000 images
Done extracting features for 38000 / 49000 images
Done extracting features for 39000 / 49000 images
Done extracting features for 40000 / 49000 images
Done extracting features for 41000 / 49000 images
Done extracting features for 42000 / 49000 images
Done extracting features for 43000 / 49000 images
Done extracting features for 44000 / 49000 images
Done extracting features for 45000 / 49000 images
Done extracting features for 46000 / 49000 images
Done extracting features for 47000 / 49000 images
Done extracting features for 48000 / 49000 images
Done extracting features for 49000 / 49000 images
```

1.3 Train Softmax classifier on features

Using the Softmax code developed earlier in the assignment, train Softmax classifiers on top of the features extracted above; this should achieve better results than training them directly on top of raw pixels.

```python
# Use the validation set to tune the learning rate and regularization strength

from icv83551.classifiers.linear_classifier import Softmax
import itertools
learning_rates = [1e-7, 1e-6, 5*1e-6, 5*1e-7]
regularization_strengths = [5e5, 5e6,1e4,5e4,5e3,5e2]

results = {}
best_val = -1
best_softmax = None

################################################################################
# TODO:                                                                        #
# Use the validation set to set the learning rate and regularization strength. #
# This should be identical to the validation that you did for the Softmax; save#
# the best trained classifer in best_softmax. If you carefully tune the model, #
# you should be able to get accuracy of above 0.42 on the validation set.      #
################################################################################
softmax = Softmax()
for lr, reg in itertools.product(
    learning_rates,  regularization_strengths
):
  softmax = Softmax()
  loss_hist = softmax.train(X_train_feats, y_train, learning_rate=lr, reg=reg,
                    num_iters=1500, verbose=True)
  y_train_pred, y_val_pred = softmax.predict(X_train_feats), softmax.
   predict(X_val_feats)
  (acc_train,acc_val) = (np.mean(y_train == y_train_pred), np.mean(y_val ==
   y_val_pred))
  results[(lr, reg)] = (acc_train,acc_val)
  if best_val < acc_val:
    best_val = acc_val
    best_softmax = softmax
# Print out results.
for lr, reg in sorted(results):
    train_accuracy, val_accuracy = results[(lr, reg)]
    print('lr %e reg %e train accuracy: %f val accuracy: %f' % (
            lr, reg, train_accuracy, val_accuracy))

print('best validation accuracy achieved: %f' % best_val)
```

```
iteration 0 / 1500: loss 845.789250
iteration 100 / 1500: loss 2.302585
iteration 200 / 1500: loss 2.302585
iteration 300 / 1500: loss 2.302585
iteration 400 / 1500: loss 2.302585
iteration 500 / 1500: loss 2.302585
iteration 600 / 1500: loss 2.302585
iteration 700 / 1500: loss 2.302585
iteration 800 / 1500: loss 2.302585
iteration 900 / 1500: loss 2.302585
iteration 1000 / 1500: loss 2.302585
iteration 1100 / 1500: loss 2.302585
iteration 1200 / 1500: loss 2.302585
iteration 1300 / 1500: loss 2.302585
iteration 1400 / 1500: loss 2.302585
iteration 0 / 1500: loss 8438.015290
iteration 100 / 1500: loss 2.302585
iteration 200 / 1500: loss 2.302585
iteration 300 / 1500: loss 2.302585
iteration 400 / 1500: loss 2.302585
iteration 500 / 1500: loss 2.302585
iteration 600 / 1500: loss 2.302585
iteration 700 / 1500: loss 2.302585
iteration 800 / 1500: loss 2.302585
iteration 900 / 1500: loss 2.302585
iteration 1000 / 1500: loss 2.302585
iteration 1100 / 1500: loss 2.302585
iteration 1200 / 1500: loss 2.302585
iteration 1300 / 1500: loss 2.302585
iteration 1400 / 1500: loss 2.302585
iteration 0 / 1500: loss 17.964297
iteration 100 / 1500: loss 12.798002
iteration 200 / 1500: loss 9.335290
iteration 300 / 1500: loss 7.015300
iteration 400 / 1500: loss 5.459599
iteration 500 / 1500: loss 4.417474
iteration 600 / 1500: loss 3.719852
iteration 700 / 1500: loss 3.252460
iteration 800 / 1500: loss 2.939209
iteration 900 / 1500: loss 2.728942
iteration 1000 / 1500: loss 2.588391
iteration 1100 / 1500: loss 2.494131
iteration 1200 / 1500: loss 2.430882
iteration 1300 / 1500: loss 2.388542
iteration 1400 / 1500: loss 2.360183
iteration 0 / 1500: loss 90.166932
iteration 100 / 1500: loss 14.074452
iteration 200 / 1500: loss 3.880022
```

```
iteration 300 / 1500: loss 2.513875
iteration 400 / 1500: loss 2.330886
iteration 500 / 1500: loss 2.306370
iteration 600 / 1500: loss 2.303088
iteration 700 / 1500: loss 2.302648
iteration 800 / 1500: loss 2.302590
iteration 900 / 1500: loss 2.302583
iteration 1000 / 1500: loss 2.302582
iteration 1100 / 1500: loss 2.302582
iteration 1200 / 1500: loss 2.302582
iteration 1300 / 1500: loss 2.302581
iteration 1400 / 1500: loss 2.302582
iteration 0 / 1500: loss 10.469632
iteration 100 / 1500: loss 8.988514
iteration 200 / 1500: loss 7.776320
iteration 300 / 1500: loss 6.783450
iteration 400 / 1500: loss 5.970116
iteration 500 / 1500: loss 5.305390
iteration 600 / 1500: loss 4.761944
iteration 700 / 1500: loss 4.314815
iteration 800 / 1500: loss 3.950408
iteration 900 / 1500: loss 3.650973
iteration 1000 / 1500: loss 3.406496
iteration 1100 / 1500: loss 3.206431
iteration 1200 / 1500: loss 3.042586
iteration 1300 / 1500: loss 2.908081
iteration 1400 / 1500: loss 2.798450
iteration 0 / 1500: loss 3.178957
iteration 100 / 1500: loss 3.161851
iteration 200 / 1500: loss 3.144924
iteration 300 / 1500: loss 3.128643
iteration 400 / 1500: loss 3.112355
iteration 500 / 1500: loss 3.094410
iteration 600 / 1500: loss 3.080498
iteration 700 / 1500: loss 3.063782
iteration 800 / 1500: loss 3.050767
iteration 900 / 1500: loss 3.035620
iteration 1000 / 1500: loss 3.021537
iteration 1100 / 1500: loss 3.007493
iteration 1200 / 1500: loss 2.991806
iteration 1300 / 1500: loss 2.978735
iteration 1400 / 1500: loss 2.964795
iteration 0 / 1500: loss 782.650757
iteration 100 / 1500: loss 2.302585
iteration 200 / 1500: loss 2.302585
iteration 300 / 1500: loss 2.302585
iteration 400 / 1500: loss 2.302585
iteration 500 / 1500: loss 2.302585
```

```
iteration 600 / 1500: loss 2.302585
iteration 700 / 1500: loss 2.302585
iteration 800 / 1500: loss 2.302585
iteration 900 / 1500: loss 2.302585
iteration 1000 / 1500: loss 2.302585
iteration 1100 / 1500: loss 2.302585
iteration 1200 / 1500: loss 2.302585
iteration 1300 / 1500: loss 2.302585
iteration 1400 / 1500: loss 2.302585
iteration 0 / 1500: loss 8808.986948

/content/drive/My
Drive/icv83551/assignments/assignment1/icv83551/classifiers/softmax.py:76:
RuntimeWarning: divide by zero encountered in log
  loss = (-np.sum(np.log(probs[np.arange(num_train), y]))/num_train) + reg*
np.sum(W * W)
/content/drive/My
Drive/icv83551/assignments/assignment1/icv83551/classifiers/softmax.py:76:
RuntimeWarning: overflow encountered in scalar multiply
  loss = (-np.sum(np.log(probs[np.arange(num_train), y]))/num_train) + reg*
np.sum(W * W)
/usr/local/lib/python3.12/dist-packages/numpy/_core/fromnumeric.py:86:
RuntimeWarning: overflow encountered in reduce
  return ufunc.reduce(obj, axis, dtype, out, **passkwargs)
/content/drive/My
Drive/icv83551/assignments/assignment1/icv83551/classifiers/softmax.py:76:
RuntimeWarning: overflow encountered in multiply
  loss = (-np.sum(np.log(probs[np.arange(num_train), y]))/num_train) + reg*
np.sum(W * W)
/content/drive/My
Drive/icv83551/assignments/assignment1/icv83551/classifiers/softmax.py:79:
RuntimeWarning: overflow encountered in multiply
  dW = X.T.dot(dscores/num_train) + 2 * reg * W

iteration 100 / 1500: loss inf
iteration 200 / 1500: loss inf
iteration 300 / 1500: loss inf
iteration 400 / 1500: loss nan
iteration 500 / 1500: loss nan
iteration 600 / 1500: loss nan
iteration 700 / 1500: loss nan
iteration 800 / 1500: loss nan
iteration 900 / 1500: loss nan
iteration 1000 / 1500: loss nan
iteration 1100 / 1500: loss nan
iteration 1200 / 1500: loss nan
iteration 1300 / 1500: loss nan
iteration 1400 / 1500: loss nan
iteration 0 / 1500: loss 19.782803
```

```
iteration 100 / 1500: loss 2.609991
iteration 200 / 1500: loss 2.307950
iteration 300 / 1500: loss 2.302669
iteration 400 / 1500: loss 2.302565
iteration 500 / 1500: loss 2.302563
iteration 600 / 1500: loss 2.302565
iteration 700 / 1500: loss 2.302564
iteration 800 / 1500: loss 2.302564
iteration 900 / 1500: loss 2.302568
iteration 1000 / 1500: loss 2.302564
iteration 1100 / 1500: loss 2.302566
iteration 1200 / 1500: loss 2.302569
iteration 1300 / 1500: loss 2.302562
iteration 1400 / 1500: loss 2.302566
iteration 0 / 1500: loss 86.538672
iteration 100 / 1500: loss 2.302582
iteration 200 / 1500: loss 2.302581
iteration 300 / 1500: loss 2.302582
iteration 400 / 1500: loss 2.302581
iteration 500 / 1500: loss 2.302581
iteration 600 / 1500: loss 2.302581
iteration 700 / 1500: loss 2.302581
iteration 800 / 1500: loss 2.302581
iteration 900 / 1500: loss 2.302581
iteration 1000 / 1500: loss 2.302582
iteration 1100 / 1500: loss 2.302581
iteration 1200 / 1500: loss 2.302582
iteration 1300 / 1500: loss 2.302582
iteration 1400 / 1500: loss 2.302581
iteration 0 / 1500: loss 10.166944
iteration 100 / 1500: loss 3.356400
iteration 200 / 1500: loss 2.443395
iteration 300 / 1500: loss 2.321475
iteration 400 / 1500: loss 2.305069
iteration 500 / 1500: loss 2.302907
iteration 600 / 1500: loss 2.302592
iteration 700 / 1500: loss 2.302552
iteration 800 / 1500: loss 2.302543
iteration 900 / 1500: loss 2.302550
iteration 1000 / 1500: loss 2.302554
iteration 1100 / 1500: loss 2.302547
iteration 1200 / 1500: loss 2.302544
iteration 1300 / 1500: loss 2.302543
iteration 1400 / 1500: loss 2.302549
iteration 0 / 1500: loss 3.166920
iteration 100 / 1500: loss 3.010754
iteration 200 / 1500: loss 2.882877
iteration 300 / 1500: loss 2.776299
```

```
iteration 400 / 1500: loss 2.691041
iteration 500 / 1500: loss 2.620503
iteration 600 / 1500: loss 2.562807
iteration 700 / 1500: loss 2.515242
iteration 800 / 1500: loss 2.476714
iteration 900 / 1500: loss 2.444852
iteration 1000 / 1500: loss 2.419323
iteration 1100 / 1500: loss 2.398712
iteration 1200 / 1500: loss 2.380135
iteration 1300 / 1500: loss 2.366347
iteration 1400 / 1500: loss 2.354910
iteration 0 / 1500: loss 848.368437
iteration 100 / 1500: loss inf
iteration 200 / 1500: loss inf
iteration 300 / 1500: loss inf
iteration 400 / 1500: loss inf
iteration 500 / 1500: loss inf
iteration 600 / 1500: loss nan
iteration 700 / 1500: loss nan
iteration 800 / 1500: loss nan
iteration 900 / 1500: loss nan
iteration 1000 / 1500: loss nan
iteration 1100 / 1500: loss nan
iteration 1200 / 1500: loss nan
iteration 1300 / 1500: loss nan
iteration 1400 / 1500: loss nan
iteration 0 / 1500: loss 8318.849549
iteration 100 / 1500: loss inf
iteration 200 / 1500: loss nan
iteration 300 / 1500: loss nan
iteration 400 / 1500: loss nan
iteration 500 / 1500: loss nan
iteration 600 / 1500: loss nan
iteration 700 / 1500: loss nan
iteration 800 / 1500: loss nan
iteration 900 / 1500: loss nan
iteration 1000 / 1500: loss nan
iteration 1100 / 1500: loss nan
iteration 1200 / 1500: loss nan
iteration 1300 / 1500: loss nan
iteration 1400 / 1500: loss nan
iteration 0 / 1500: loss 19.407281
iteration 100 / 1500: loss 2.302564
iteration 200 / 1500: loss 2.302571
iteration 300 / 1500: loss 2.302566
iteration 400 / 1500: loss 2.302569
iteration 500 / 1500: loss 2.302567
iteration 600 / 1500: loss 2.302565
```

```
iteration 700 / 1500: loss 2.302572
iteration 800 / 1500: loss 2.302561
iteration 900 / 1500: loss 2.302562
iteration 1000 / 1500: loss 2.302566
iteration 1100 / 1500: loss 2.302564
iteration 1200 / 1500: loss 2.302565
iteration 1300 / 1500: loss 2.302569
iteration 1400 / 1500: loss 2.302568
iteration 0 / 1500: loss 87.127238
iteration 100 / 1500: loss 2.302582
iteration 200 / 1500: loss 2.302583
iteration 300 / 1500: loss 2.302583
iteration 400 / 1500: loss 2.302583
iteration 500 / 1500: loss 2.302583
iteration 600 / 1500: loss 2.302582
iteration 700 / 1500: loss 2.302583
iteration 800 / 1500: loss 2.302582
iteration 900 / 1500: loss 2.302583
iteration 1000 / 1500: loss 2.302582
iteration 1100 / 1500: loss 2.302583
iteration 1200 / 1500: loss 2.302583
iteration 1300 / 1500: loss 2.302582
iteration 1400 / 1500: loss 2.302583
iteration 0 / 1500: loss 10.517585
iteration 100 / 1500: loss 2.302844
iteration 200 / 1500: loss 2.302553
iteration 300 / 1500: loss 2.302550
iteration 400 / 1500: loss 2.302543
iteration 500 / 1500: loss 2.302561
iteration 600 / 1500: loss 2.302544
iteration 700 / 1500: loss 2.302547
iteration 800 / 1500: loss 2.302555
iteration 900 / 1500: loss 2.302547
iteration 1000 / 1500: loss 2.302546
iteration 1100 / 1500: loss 2.302543
iteration 1200 / 1500: loss 2.302555
iteration 1300 / 1500: loss 2.302545
iteration 1400 / 1500: loss 2.302553
iteration 0 / 1500: loss 3.150441
iteration 100 / 1500: loss 2.613700
iteration 200 / 1500: loss 2.416441
iteration 300 / 1500: loss 2.344283
iteration 400 / 1500: loss 2.317563
iteration 500 / 1500: loss 2.307854
iteration 600 / 1500: loss 2.304262
iteration 700 / 1500: loss 2.302935
iteration 800 / 1500: loss 2.302564
iteration 900 / 1500: loss 2.302340
```

```
iteration 1000 / 1500: loss 2.302293
iteration 1100 / 1500: loss 2.302269
iteration 1200 / 1500: loss 2.302248
iteration 1300 / 1500: loss 2.302143
iteration 1400 / 1500: loss 2.302193
iteration 0 / 1500: loss 886.330450
iteration 100 / 1500: loss 2.302585
iteration 200 / 1500: loss 2.302585
iteration 300 / 1500: loss 2.302585
iteration 400 / 1500: loss 2.302585
iteration 500 / 1500: loss 2.302585
iteration 600 / 1500: loss 2.302585
iteration 700 / 1500: loss 2.302585
iteration 800 / 1500: loss 2.302585
iteration 900 / 1500: loss 2.302585
iteration 1000 / 1500: loss 2.302585
iteration 1100 / 1500: loss 2.302585
iteration 1200 / 1500: loss 2.302585
iteration 1300 / 1500: loss 2.302585
iteration 1400 / 1500: loss 2.302585
iteration 0 / 1500: loss 7915.086771
iteration 100 / 1500: loss inf
iteration 200 / 1500: loss inf
iteration 300 / 1500: loss inf
iteration 400 / 1500: loss inf
iteration 500 / 1500: loss inf
iteration 600 / 1500: loss nan
iteration 700 / 1500: loss nan
iteration 800 / 1500: loss nan
iteration 900 / 1500: loss nan
iteration 1000 / 1500: loss nan
iteration 1100 / 1500: loss nan
iteration 1200 / 1500: loss nan
iteration 1300 / 1500: loss nan
iteration 1400 / 1500: loss nan
iteration 0 / 1500: loss 18.967198
iteration 100 / 1500: loss 4.535115
iteration 200 / 1500: loss 2.601703
iteration 300 / 1500: loss 2.342661
iteration 400 / 1500: loss 2.307956
iteration 500 / 1500: loss 2.303284
iteration 600 / 1500: loss 2.302667
iteration 700 / 1500: loss 2.302579
iteration 800 / 1500: loss 2.302570
iteration 900 / 1500: loss 2.302567
iteration 1000 / 1500: loss 2.302567
iteration 1100 / 1500: loss 2.302569
iteration 1200 / 1500: loss 2.302563
```

```
iteration 1300 / 1500: loss 2.302565
iteration 1400 / 1500: loss 2.302568
iteration 0 / 1500: loss 91.589526
iteration 100 / 1500: loss 2.305720
iteration 200 / 1500: loss 2.302581
iteration 300 / 1500: loss 2.302582
iteration 400 / 1500: loss 2.302581
iteration 500 / 1500: loss 2.302581
iteration 600 / 1500: loss 2.302581
iteration 700 / 1500: loss 2.302581
iteration 800 / 1500: loss 2.302581
iteration 900 / 1500: loss 2.302581
iteration 1000 / 1500: loss 2.302580
iteration 1100 / 1500: loss 2.302582
iteration 1200 / 1500: loss 2.302581
iteration 1300 / 1500: loss 2.302582
iteration 1400 / 1500: loss 2.302583
iteration 0 / 1500: loss 10.704427
iteration 100 / 1500: loss 5.385509
iteration 200 / 1500: loss 3.433733
iteration 300 / 1500: loss 2.717924
iteration 400 / 1500: loss 2.454895
iteration 500 / 1500: loss 2.358410
iteration 600 / 1500: loss 2.323050
iteration 700 / 1500: loss 2.310077
iteration 800 / 1500: loss 2.305271
iteration 900 / 1500: loss 2.303554
iteration 1000 / 1500: loss 2.302930
iteration 1100 / 1500: loss 2.302681
iteration 1200 / 1500: loss 2.302600
iteration 1300 / 1500: loss 2.302558
iteration 1400 / 1500: loss 2.302551
iteration 0 / 1500: loss 3.140639
iteration 100 / 1500: loss 3.061507
iteration 200 / 1500: loss 2.988288
iteration 300 / 1500: loss 2.925170
iteration 400 / 1500: loss 2.864783
iteration 500 / 1500: loss 2.811449
iteration 600 / 1500: loss 2.761774
iteration 700 / 1500: loss 2.717615
iteration 800 / 1500: loss 2.678492
iteration 900 / 1500: loss 2.642882
iteration 1000 / 1500: loss 2.611134
iteration 1100 / 1500: loss 2.580598
iteration 1200 / 1500: loss 2.554259
iteration 1300 / 1500: loss 2.530078
iteration 1400 / 1500: loss 2.509441
lr 1.000000e-07 reg 5.000000e+02 train accuracy: 0.109429 val accuracy: 0.120000
```

```
lr 1.000000e-07 reg 5.000000e+03 train accuracy: 0.110551 val accuracy: 0.104000
lr 1.000000e-07 reg 1.000000e+04 train accuracy: 0.108469 val accuracy: 0.098000
lr 1.000000e-07 reg 5.000000e+04 train accuracy: 0.419224 val accuracy: 0.409000
lr 1.000000e-07 reg 5.000000e+05 train accuracy: 0.418449 val accuracy: 0.425000
lr 1.000000e-07 reg 5.000000e+06 train accuracy: 0.331959 val accuracy: 0.339000
lr 5.000000e-07 reg 5.000000e+02 train accuracy: 0.119755 val accuracy: 0.111000
lr 5.000000e-07 reg 5.000000e+03 train accuracy: 0.419204 val accuracy: 0.416000
lr 5.000000e-07 reg 1.000000e+04 train accuracy: 0.423449 val accuracy: 0.416000
lr 5.000000e-07 reg 5.000000e+04 train accuracy: 0.420837 val accuracy: 0.423000
lr 5.000000e-07 reg 5.000000e+05 train accuracy: 0.374939 val accuracy: 0.361000
lr 5.000000e-07 reg 5.000000e+06 train accuracy: 0.100265 val accuracy: 0.087000
lr 1.000000e-06 reg 5.000000e+02 train accuracy: 0.141184 val accuracy: 0.144000
lr 1.000000e-06 reg 5.000000e+03 train accuracy: 0.420286 val accuracy: 0.427000
lr 1.000000e-06 reg 1.000000e+04 train accuracy: 0.415347 val accuracy: 0.402000
lr 1.000000e-06 reg 5.000000e+04 train accuracy: 0.405980 val accuracy: 0.386000
lr 1.000000e-06 reg 5.000000e+05 train accuracy: 0.342673 val accuracy: 0.337000
lr 1.000000e-06 reg 5.000000e+06 train accuracy: 0.100265 val accuracy: 0.087000
lr 5.000000e-06 reg 5.000000e+02 train accuracy: 0.423959 val accuracy: 0.433000
lr 5.000000e-06 reg 5.000000e+03 train accuracy: 0.418122 val accuracy: 0.411000
lr 5.000000e-06 reg 1.000000e+04 train accuracy: 0.414714 val accuracy: 0.408000
lr 5.000000e-06 reg 5.000000e+04 train accuracy: 0.371388 val accuracy: 0.355000
lr 5.000000e-06 reg 5.000000e+05 train accuracy: 0.100265 val accuracy: 0.087000
lr 5.000000e-06 reg 5.000000e+06 train accuracy: 0.100265 val accuracy: 0.087000
best validation accuracy achieved: 0.433000
```

[6]:
```python
# Evaluate your trained Softmax on the test set: you should be able to get at
 ↪least 0.42
y_test_pred = best_softmax.predict(X_test_feats)
test_accuracy = np.mean(y_test == y_test_pred)
print(test_accuracy)
```

```
0.421
```

[7]:
```python
# Save best softmax model
best_softmax.save("best_softmax_features.npy")
```

```
best_softmax_features.npy saved.
```

[8]:
```python
# An important way to gain intuition about how an algorithm works is to
# visualize the mistakes that it makes. In this visualization, we show examples
# of images that are misclassified by our current system. The first column
# shows images that our system labeled as "plane" but whose true label is
# something other than "plane".

examples_per_class = 8
classes = ['plane', 'car', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse',
 ↪'ship', 'truck']
```

```python
for cls, cls_name in enumerate(classes):
    idxs = np.where((y_test != cls) & (y_test_pred == cls))[0]
    idxs = np.random.choice(idxs, examples_per_class, replace=False)
    for i, idx in enumerate(idxs):
        plt.subplot(examples_per_class, len(classes), i * len(classes) + cls +␣
 ↪1)
        plt.imshow(X_test[idx].astype('uint8'))
        plt.axis('off')
        if i == 0:
            plt.title(cls_name)
plt.show()
```
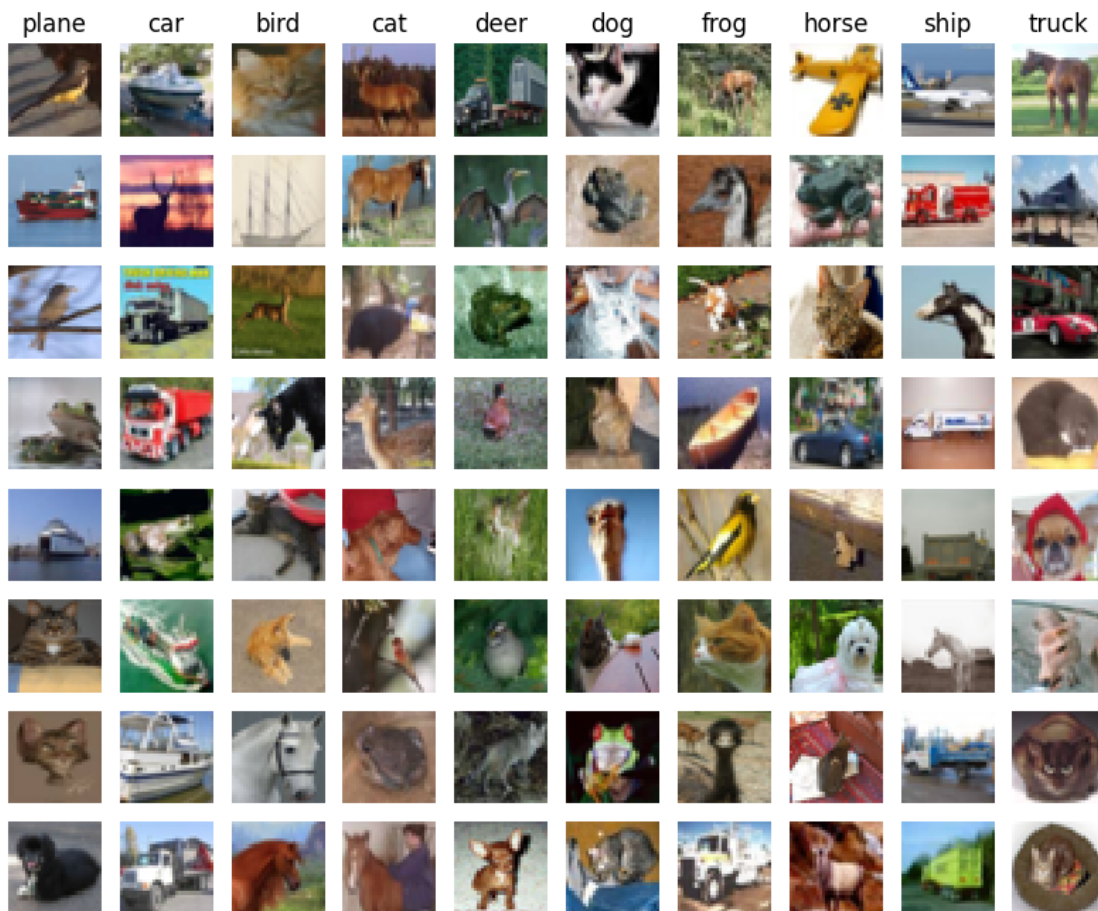

1.3.1 Inline question 1:

Describe the misclassification results that you see. Do they make sense?

Your Answer : we can see that there are many visually similar classes, meaning that the model misclassifies them , we can also see that the correct precentage was: 0.417, which is much higher then a random 0.1 classifier. but since this is a linear classifier, if still cant take into account more

15

complex features (direction of pic, varaitains ect), so it makes sense that these are the results (linear classifiers are very color based, less object base)

1.4 Neural Network on image features

Earlier in this assigment we saw that training a two-layer neural network on raw pixels achieved better classification performance than linear classifiers on raw pixels. In this notebook we have seen that linear classifiers on image features outperform linear classifiers on raw pixels.

For completeness, we should also try training a neural network on image features. This approach should outperform all previous approaches: you should easily be able to achieve over 55% classification accuracy on the test set; our best model achieves about 60% classification accuracy.

```python
# Preprocessing: Remove the bias dimension
# Make sure to run this cell only ONCE
print(X_train_feats.shape)
X_train_feats = X_train_feats[:, :-1]
X_val_feats = X_val_feats[:, :-1]
X_test_feats = X_test_feats[:, :-1]


print(X_train_feats.shape)
```

```
(49000, 170)
(49000, 169)
```

```python
from icv83551.classifiers.fc_net import TwoLayerNet
from icv83551.solver import Solver

input_dim = X_train_feats.shape[1]
hidden_dim = 500
num_classes = 10
better_scale = np.sqrt(2.0 / input_dim)

# Adjusting hyperparameters to more reasonable ranges
learningRate = [0.05,0.01, 0.03]
reg_st =[0.99,1, 0.98]
batch_size = [200]
data = {
    'X_train': X_train_feats,
    'y_train': y_train,
    'X_val': X_val_feats,
    'y_val': y_val,
    'X_test': X_test_feats,
    'y_test': y_test,
}
best_net = None
highest_acc = -1
```

```python
for lr, reg_s in itertools.product(
    learningRate, reg_st
):
  model = TwoLayerNet(input_dim, hidden_dim, num_classes,weight_scale=␣
  ↪better_scale, reg=reg_s)
  #model = TwoLayerNet(input_dim, hidden_dim, num_classes, reg=reg_s)
  solver = Solver(model,
                  data,
                  update_rule='sgd',
                  optim_config={'learning_rate': lr},
                  lr_decay=0.98,
                  num_epochs=200,
                  batch_size=200,
                  print_every=100)
  solver.train()
  acc = solver.check_accuracy(data['X_val'], data['y_val'])
  if acc > highest_acc:
    highest_acc = acc
    best_net = model
```

Streaming output truncated to the last 5000 lines.
(Epoch 153 / 200) train acc: 0.089000; val_acc: 0.079000
(Iteration 37501 / 49000) loss: 2.302696
(Iteration 37601 / 49000) loss: 2.302485
(Iteration 37701 / 49000) loss: 2.302548
(Epoch 154 / 200) train acc: 0.080000; val_acc: 0.079000
(Iteration 37801 / 49000) loss: 2.302903
(Iteration 37901 / 49000) loss: 2.302676
(Epoch 155 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 38001 / 49000) loss: 2.302712
(Iteration 38101 / 49000) loss: 2.302661
(Iteration 38201 / 49000) loss: 2.302669
(Epoch 156 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 38301 / 49000) loss: 2.302955
(Iteration 38401 / 49000) loss: 2.302592
(Epoch 157 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 38501 / 49000) loss: 2.302875
(Iteration 38601 / 49000) loss: 2.302651
(Iteration 38701 / 49000) loss: 2.302623
(Epoch 158 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 38801 / 49000) loss: 2.302586
(Iteration 38901 / 49000) loss: 2.303353
(Epoch 159 / 200) train acc: 0.117000; val_acc: 0.078000
(Iteration 39001 / 49000) loss: 2.302460
(Iteration 39101 / 49000) loss: 2.302642
(Epoch 160 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 39201 / 49000) loss: 2.302510

17

```
(Iteration 39301 / 49000) loss: 2.302790
(Iteration 39401 / 49000) loss: 2.302755
(Epoch 161 / 200) train acc: 0.085000; val_acc: 0.078000
(Iteration 39501 / 49000) loss: 2.302885
(Iteration 39601 / 49000) loss: 2.302491
(Epoch 162 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 39701 / 49000) loss: 2.302847
(Iteration 39801 / 49000) loss: 2.302808
(Iteration 39901 / 49000) loss: 2.302656
(Epoch 163 / 200) train acc: 0.113000; val_acc: 0.078000
(Iteration 40001 / 49000) loss: 2.302330
(Iteration 40101 / 49000) loss: 2.302545
(Epoch 164 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 40201 / 49000) loss: 2.302749
(Iteration 40301 / 49000) loss: 2.302535
(Iteration 40401 / 49000) loss: 2.302631
(Epoch 165 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 40501 / 49000) loss: 2.302288
(Iteration 40601 / 49000) loss: 2.302848
(Epoch 166 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 40701 / 49000) loss: 2.302335
(Iteration 40801 / 49000) loss: 2.302181
(Iteration 40901 / 49000) loss: 2.302384
(Epoch 167 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 41001 / 49000) loss: 2.302640
(Iteration 41101 / 49000) loss: 2.302394
(Epoch 168 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 41201 / 49000) loss: 2.302515
(Iteration 41301 / 49000) loss: 2.302701
(Iteration 41401 / 49000) loss: 2.302899
(Epoch 169 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 41501 / 49000) loss: 2.303083
(Iteration 41601 / 49000) loss: 2.302637
(Epoch 170 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 41701 / 49000) loss: 2.302559
(Iteration 41801 / 49000) loss: 2.302962
(Epoch 171 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 41901 / 49000) loss: 2.302454
(Iteration 42001 / 49000) loss: 2.303034
(Iteration 42101 / 49000) loss: 2.302628
(Epoch 172 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 42201 / 49000) loss: 2.302352
(Iteration 42301 / 49000) loss: 2.302835
(Epoch 173 / 200) train acc: 0.121000; val_acc: 0.079000
(Iteration 42401 / 49000) loss: 2.302535
(Iteration 42501 / 49000) loss: 2.302709
(Iteration 42601 / 49000) loss: 2.302683
(Epoch 174 / 200) train acc: 0.089000; val_acc: 0.078000
```

```
(Iteration 42701 / 49000) loss: 2.302521
(Iteration 42801 / 49000) loss: 2.302610
(Epoch 175 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 42901 / 49000) loss: 2.302518
(Iteration 43001 / 49000) loss: 2.302838
(Iteration 43101 / 49000) loss: 2.302530
(Epoch 176 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 43201 / 49000) loss: 2.302572
(Iteration 43301 / 49000) loss: 2.302588
(Epoch 177 / 200) train acc: 0.111000; val_acc: 0.079000
(Iteration 43401 / 49000) loss: 2.302576
(Iteration 43501 / 49000) loss: 2.302891
(Iteration 43601 / 49000) loss: 2.302451
(Epoch 178 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 43701 / 49000) loss: 2.302819
(Iteration 43801 / 49000) loss: 2.302508
(Epoch 179 / 200) train acc: 0.085000; val_acc: 0.079000
(Iteration 43901 / 49000) loss: 2.302354
(Iteration 44001 / 49000) loss: 2.302865
(Epoch 180 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 44101 / 49000) loss: 2.302527
(Iteration 44201 / 49000) loss: 2.302475
(Iteration 44301 / 49000) loss: 2.302561
(Epoch 181 / 200) train acc: 0.113000; val_acc: 0.079000
(Iteration 44401 / 49000) loss: 2.302502
(Iteration 44501 / 49000) loss: 2.302446
(Epoch 182 / 200) train acc: 0.111000; val_acc: 0.079000
(Iteration 44601 / 49000) loss: 2.302812
(Iteration 44701 / 49000) loss: 2.302470
(Iteration 44801 / 49000) loss: 2.302929
(Epoch 183 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 44901 / 49000) loss: 2.302757
(Iteration 45001 / 49000) loss: 2.302300
(Epoch 184 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 45101 / 49000) loss: 2.302724
(Iteration 45201 / 49000) loss: 2.302368
(Iteration 45301 / 49000) loss: 2.302441
(Epoch 185 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 45401 / 49000) loss: 2.302586
(Iteration 45501 / 49000) loss: 2.302271
(Epoch 186 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 45601 / 49000) loss: 2.302862
(Iteration 45701 / 49000) loss: 2.302788
(Iteration 45801 / 49000) loss: 2.302773
(Epoch 187 / 200) train acc: 0.089000; val_acc: 0.079000
(Iteration 45901 / 49000) loss: 2.302260
(Iteration 46001 / 49000) loss: 2.302646
(Epoch 188 / 200) train acc: 0.103000; val_acc: 0.079000
```

```
(Iteration 46101 / 49000) loss: 2.303119
(Iteration 46201 / 49000) loss: 2.302884
(Iteration 46301 / 49000) loss: 2.302506
(Epoch 189 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 46401 / 49000) loss: 2.302531
(Iteration 46501 / 49000) loss: 2.302643
(Epoch 190 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 46601 / 49000) loss: 2.302491
(Iteration 46701 / 49000) loss: 2.302513
(Epoch 191 / 200) train acc: 0.086000; val_acc: 0.079000
(Iteration 46801 / 49000) loss: 2.302643
(Iteration 46901 / 49000) loss: 2.302442
(Iteration 47001 / 49000) loss: 2.302123
(Epoch 192 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 47101 / 49000) loss: 2.302351
(Iteration 47201 / 49000) loss: 2.302334
(Epoch 193 / 200) train acc: 0.119000; val_acc: 0.078000
(Iteration 47301 / 49000) loss: 2.302450
(Iteration 47401 / 49000) loss: 2.302378
(Iteration 47501 / 49000) loss: 2.302437
(Epoch 194 / 200) train acc: 0.089000; val_acc: 0.079000
(Iteration 47601 / 49000) loss: 2.302651
(Iteration 47701 / 49000) loss: 2.302737
(Epoch 195 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 47801 / 49000) loss: 2.302547
(Iteration 47901 / 49000) loss: 2.302478
(Iteration 48001 / 49000) loss: 2.302681
(Epoch 196 / 200) train acc: 0.082000; val_acc: 0.079000
(Iteration 48101 / 49000) loss: 2.302423
(Iteration 48201 / 49000) loss: 2.302862
(Epoch 197 / 200) train acc: 0.115000; val_acc: 0.079000
(Iteration 48301 / 49000) loss: 2.302655
(Iteration 48401 / 49000) loss: 2.302595
(Iteration 48501 / 49000) loss: 2.302734
(Epoch 198 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 48601 / 49000) loss: 2.302659
(Iteration 48701 / 49000) loss: 2.302358
(Epoch 199 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 48801 / 49000) loss: 2.302567
(Iteration 48901 / 49000) loss: 2.302416
(Epoch 200 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 1 / 49000) loss: 521.104268
(Epoch 0 / 200) train acc: 0.123000; val_acc: 0.115000
(Iteration 101 / 49000) loss: 2.326748
(Iteration 201 / 49000) loss: 2.303569
(Epoch 1 / 200) train acc: 0.098000; val_acc: 0.107000
(Iteration 301 / 49000) loss: 2.302989
(Iteration 401 / 49000) loss: 2.302863
```

```
(Epoch 2 / 200) train acc: 0.110000; val_acc: 0.107000
(Iteration 501 / 49000) loss: 2.303478
(Iteration 601 / 49000) loss: 2.302799
(Iteration 701 / 49000) loss: 2.302135
(Epoch 3 / 200) train acc: 0.100000; val_acc: 0.112000
(Iteration 801 / 49000) loss: 2.302143
(Iteration 901 / 49000) loss: 2.303748
(Epoch 4 / 200) train acc: 0.101000; val_acc: 0.105000
(Iteration 1001 / 49000) loss: 2.302825
(Iteration 1101 / 49000) loss: 2.301275
(Iteration 1201 / 49000) loss: 2.302796
(Epoch 5 / 200) train acc: 0.113000; val_acc: 0.079000
(Iteration 1301 / 49000) loss: 2.302209
(Iteration 1401 / 49000) loss: 2.302703
(Epoch 6 / 200) train acc: 0.089000; val_acc: 0.087000
(Iteration 1501 / 49000) loss: 2.301548
(Iteration 1601 / 49000) loss: 2.304221
(Iteration 1701 / 49000) loss: 2.302953
(Epoch 7 / 200) train acc: 0.097000; val_acc: 0.102000
(Iteration 1801 / 49000) loss: 2.301788
(Iteration 1901 / 49000) loss: 2.303001
(Epoch 8 / 200) train acc: 0.116000; val_acc: 0.119000
(Iteration 2001 / 49000) loss: 2.302223
(Iteration 2101 / 49000) loss: 2.302849
(Iteration 2201 / 49000) loss: 2.302914
(Epoch 9 / 200) train acc: 0.111000; val_acc: 0.078000
(Iteration 2301 / 49000) loss: 2.303549
(Iteration 2401 / 49000) loss: 2.303453
(Epoch 10 / 200) train acc: 0.095000; val_acc: 0.078000
(Iteration 2501 / 49000) loss: 2.302005
(Iteration 2601 / 49000) loss: 2.303760
(Epoch 11 / 200) train acc: 0.082000; val_acc: 0.078000
(Iteration 2701 / 49000) loss: 2.301755
(Iteration 2801 / 49000) loss: 2.303237
(Iteration 2901 / 49000) loss: 2.302782
(Epoch 12 / 200) train acc: 0.110000; val_acc: 0.087000
(Iteration 3001 / 49000) loss: 2.303049
(Iteration 3101 / 49000) loss: 2.301655
(Epoch 13 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 3201 / 49000) loss: 2.302371
(Iteration 3301 / 49000) loss: 2.302646
(Iteration 3401 / 49000) loss: 2.302068
(Epoch 14 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 3501 / 49000) loss: 2.302627
(Iteration 3601 / 49000) loss: 2.302236
(Epoch 15 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 3701 / 49000) loss: 2.301786
(Iteration 3801 / 49000) loss: 2.301438
```

```
(Iteration 3901 / 49000) loss: 2.302032
(Epoch 16 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 4001 / 49000) loss: 2.302954
(Iteration 4101 / 49000) loss: 2.302824
(Epoch 17 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 4201 / 49000) loss: 2.302542
(Iteration 4301 / 49000) loss: 2.302849
(Iteration 4401 / 49000) loss: 2.302250
(Epoch 18 / 200) train acc: 0.111000; val_acc: 0.079000
(Iteration 4501 / 49000) loss: 2.303049
(Iteration 4601 / 49000) loss: 2.301925
(Epoch 19 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 4701 / 49000) loss: 2.301973
(Iteration 4801 / 49000) loss: 2.302842
(Epoch 20 / 200) train acc: 0.088000; val_acc: 0.079000
(Iteration 4901 / 49000) loss: 2.302914
(Iteration 5001 / 49000) loss: 2.302665
(Iteration 5101 / 49000) loss: 2.302973
(Epoch 21 / 200) train acc: 0.110000; val_acc: 0.107000
(Iteration 5201 / 49000) loss: 2.302552
(Iteration 5301 / 49000) loss: 2.302735
(Epoch 22 / 200) train acc: 0.122000; val_acc: 0.105000
(Iteration 5401 / 49000) loss: 2.303043
(Iteration 5501 / 49000) loss: 2.302262
(Iteration 5601 / 49000) loss: 2.302496
(Epoch 23 / 200) train acc: 0.086000; val_acc: 0.112000
(Iteration 5701 / 49000) loss: 2.302698
(Iteration 5801 / 49000) loss: 2.302667
(Epoch 24 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 5901 / 49000) loss: 2.302290
(Iteration 6001 / 49000) loss: 2.302927
(Iteration 6101 / 49000) loss: 2.303421
(Epoch 25 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 6201 / 49000) loss: 2.302922
(Iteration 6301 / 49000) loss: 2.302166
(Epoch 26 / 200) train acc: 0.095000; val_acc: 0.098000
(Iteration 6401 / 49000) loss: 2.304025
(Iteration 6501 / 49000) loss: 2.301770
(Iteration 6601 / 49000) loss: 2.303526
(Epoch 27 / 200) train acc: 0.094000; val_acc: 0.102000
(Iteration 6701 / 49000) loss: 2.302612
(Iteration 6801 / 49000) loss: 2.301682
(Epoch 28 / 200) train acc: 0.113000; val_acc: 0.078000
(Iteration 6901 / 49000) loss: 2.304659
(Iteration 7001 / 49000) loss: 2.302208
(Iteration 7101 / 49000) loss: 2.303397
(Epoch 29 / 200) train acc: 0.088000; val_acc: 0.098000
(Iteration 7201 / 49000) loss: 2.301319
```

```
(Iteration 7301 / 49000) loss: 2.302712
(Epoch 30 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 7401 / 49000) loss: 2.302922
(Iteration 7501 / 49000) loss: 2.303280
(Epoch 31 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 7601 / 49000) loss: 2.302423
(Iteration 7701 / 49000) loss: 2.302743
(Iteration 7801 / 49000) loss: 2.303177
(Epoch 32 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 7901 / 49000) loss: 2.301947
(Iteration 8001 / 49000) loss: 2.303371
(Epoch 33 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 8101 / 49000) loss: 2.302146
(Iteration 8201 / 49000) loss: 2.302889
(Iteration 8301 / 49000) loss: 2.302802
(Epoch 34 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 8401 / 49000) loss: 2.303247
(Iteration 8501 / 49000) loss: 2.302091
(Epoch 35 / 200) train acc: 0.112000; val_acc: 0.112000
(Iteration 8601 / 49000) loss: 2.302671
(Iteration 8701 / 49000) loss: 2.302706
(Iteration 8801 / 49000) loss: 2.301950
(Epoch 36 / 200) train acc: 0.088000; val_acc: 0.098000
(Iteration 8901 / 49000) loss: 2.301667
(Iteration 9001 / 49000) loss: 2.302465
(Epoch 37 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 9101 / 49000) loss: 2.303083
(Iteration 9201 / 49000) loss: 2.302367
(Iteration 9301 / 49000) loss: 2.302196
(Epoch 38 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 9401 / 49000) loss: 2.302809
(Iteration 9501 / 49000) loss: 2.302721
(Epoch 39 / 200) train acc: 0.121000; val_acc: 0.087000
(Iteration 9601 / 49000) loss: 2.302312
(Iteration 9701 / 49000) loss: 2.303034
(Epoch 40 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 9801 / 49000) loss: 2.301795
(Iteration 9901 / 49000) loss: 2.303035
(Iteration 10001 / 49000) loss: 2.301950
(Epoch 41 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 10101 / 49000) loss: 2.303271
(Iteration 10201 / 49000) loss: 2.301664
(Epoch 42 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 10301 / 49000) loss: 2.302467
(Iteration 10401 / 49000) loss: 2.302043
(Iteration 10501 / 49000) loss: 2.302821
(Epoch 43 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 10601 / 49000) loss: 2.302467
```

```
(Iteration 10701 / 49000) loss: 2.302095
(Epoch 44 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 10801 / 49000) loss: 2.302595
(Iteration 10901 / 49000) loss: 2.301319
(Iteration 11001 / 49000) loss: 2.302597
(Epoch 45 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 11101 / 49000) loss: 2.302108
(Iteration 11201 / 49000) loss: 2.301217
(Epoch 46 / 200) train acc: 0.089000; val_acc: 0.079000
(Iteration 11301 / 49000) loss: 2.302955
(Iteration 11401 / 49000) loss: 2.302426
(Iteration 11501 / 49000) loss: 2.302005
(Epoch 47 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 11601 / 49000) loss: 2.302503
(Iteration 11701 / 49000) loss: 2.301835
(Epoch 48 / 200) train acc: 0.078000; val_acc: 0.079000
(Iteration 11801 / 49000) loss: 2.301929
(Iteration 11901 / 49000) loss: 2.303334
(Iteration 12001 / 49000) loss: 2.303026
(Epoch 49 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 12101 / 49000) loss: 2.302467
(Iteration 12201 / 49000) loss: 2.302916
(Epoch 50 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 12301 / 49000) loss: 2.303917
(Iteration 12401 / 49000) loss: 2.302342
(Epoch 51 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 12501 / 49000) loss: 2.302439
(Iteration 12601 / 49000) loss: 2.302572
(Iteration 12701 / 49000) loss: 2.303208
(Epoch 52 / 200) train acc: 0.082000; val_acc: 0.079000
(Iteration 12801 / 49000) loss: 2.302883
(Iteration 12901 / 49000) loss: 2.302052
(Epoch 53 / 200) train acc: 0.120000; val_acc: 0.079000
(Iteration 13001 / 49000) loss: 2.302558
(Iteration 13101 / 49000) loss: 2.303120
(Iteration 13201 / 49000) loss: 2.302265
(Epoch 54 / 200) train acc: 0.110000; val_acc: 0.079000
(Iteration 13301 / 49000) loss: 2.304221
(Iteration 13401 / 49000) loss: 2.302914
(Epoch 55 / 200) train acc: 0.101000; val_acc: 0.119000
(Iteration 13501 / 49000) loss: 2.301873
(Iteration 13601 / 49000) loss: 2.302701
(Iteration 13701 / 49000) loss: 2.302200
(Epoch 56 / 200) train acc: 0.096000; val_acc: 0.119000
(Iteration 13801 / 49000) loss: 2.302662
(Iteration 13901 / 49000) loss: 2.303086
(Epoch 57 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 14001 / 49000) loss: 2.301616
```

```
(Iteration 14101 / 49000) loss: 2.302083
(Iteration 14201 / 49000) loss: 2.302037
(Epoch 58 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 14301 / 49000) loss: 2.302182
(Iteration 14401 / 49000) loss: 2.301802
(Epoch 59 / 200) train acc: 0.079000; val_acc: 0.079000
(Iteration 14501 / 49000) loss: 2.301592
(Iteration 14601 / 49000) loss: 2.302758
(Epoch 60 / 200) train acc: 0.092000; val_acc: 0.098000
(Iteration 14701 / 49000) loss: 2.302723
(Iteration 14801 / 49000) loss: 2.302786
(Iteration 14901 / 49000) loss: 2.302468
(Epoch 61 / 200) train acc: 0.093000; val_acc: 0.098000
(Iteration 15001 / 49000) loss: 2.302734
(Iteration 15101 / 49000) loss: 2.303037
(Epoch 62 / 200) train acc: 0.110000; val_acc: 0.087000
(Iteration 15201 / 49000) loss: 2.302201
(Iteration 15301 / 49000) loss: 2.302228
(Iteration 15401 / 49000) loss: 2.302542
(Epoch 63 / 200) train acc: 0.106000; val_acc: 0.102000
(Iteration 15501 / 49000) loss: 2.303317
(Iteration 15601 / 49000) loss: 2.302697
(Epoch 64 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 15701 / 49000) loss: 2.302178
(Iteration 15801 / 49000) loss: 2.302558
(Iteration 15901 / 49000) loss: 2.302952
(Epoch 65 / 200) train acc: 0.114000; val_acc: 0.087000
(Iteration 16001 / 49000) loss: 2.302818
(Iteration 16101 / 49000) loss: 2.302712
(Epoch 66 / 200) train acc: 0.101000; val_acc: 0.098000
(Iteration 16201 / 49000) loss: 2.302109
(Iteration 16301 / 49000) loss: 2.302618
(Iteration 16401 / 49000) loss: 2.302383
(Epoch 67 / 200) train acc: 0.090000; val_acc: 0.113000
(Iteration 16501 / 49000) loss: 2.302470
(Iteration 16601 / 49000) loss: 2.302550
(Epoch 68 / 200) train acc: 0.106000; val_acc: 0.113000
(Iteration 16701 / 49000) loss: 2.302880
(Iteration 16801 / 49000) loss: 2.302872
(Iteration 16901 / 49000) loss: 2.302815
(Epoch 69 / 200) train acc: 0.101000; val_acc: 0.113000
(Iteration 17001 / 49000) loss: 2.302649
(Iteration 17101 / 49000) loss: 2.302535
(Epoch 70 / 200) train acc: 0.101000; val_acc: 0.105000
(Iteration 17201 / 49000) loss: 2.302867
(Iteration 17301 / 49000) loss: 2.301852
(Epoch 71 / 200) train acc: 0.095000; val_acc: 0.087000
(Iteration 17401 / 49000) loss: 2.302300
```

```
(Iteration 17501 / 49000) loss: 2.302710
(Iteration 17601 / 49000) loss: 2.302556
(Epoch 72 / 200) train acc: 0.096000; val_acc: 0.113000
(Iteration 17701 / 49000) loss: 2.302732
(Iteration 17801 / 49000) loss: 2.302775
(Epoch 73 / 200) train acc: 0.111000; val_acc: 0.078000
(Iteration 17901 / 49000) loss: 2.301844
(Iteration 18001 / 49000) loss: 2.302257
(Iteration 18101 / 49000) loss: 2.303155
(Epoch 74 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 18201 / 49000) loss: 2.302896
(Iteration 18301 / 49000) loss: 2.302210
(Epoch 75 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 18401 / 49000) loss: 2.302229
(Iteration 18501 / 49000) loss: 2.302909
(Iteration 18601 / 49000) loss: 2.302692
(Epoch 76 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 18701 / 49000) loss: 2.302096
(Iteration 18801 / 49000) loss: 2.303063
(Epoch 77 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 18901 / 49000) loss: 2.302023
(Iteration 19001 / 49000) loss: 2.302918
(Iteration 19101 / 49000) loss: 2.302418
(Epoch 78 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 19201 / 49000) loss: 2.302695
(Iteration 19301 / 49000) loss: 2.302597
(Epoch 79 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 19401 / 49000) loss: 2.302627
(Iteration 19501 / 49000) loss: 2.302936
(Epoch 80 / 200) train acc: 0.101000; val_acc: 0.112000
(Iteration 19601 / 49000) loss: 2.302585
(Iteration 19701 / 49000) loss: 2.303110
(Iteration 19801 / 49000) loss: 2.302271
(Epoch 81 / 200) train acc: 0.101000; val_acc: 0.102000
(Iteration 19901 / 49000) loss: 2.302824
(Iteration 20001 / 49000) loss: 2.302449
(Epoch 82 / 200) train acc: 0.107000; val_acc: 0.102000
(Iteration 20101 / 49000) loss: 2.302386
(Iteration 20201 / 49000) loss: 2.303045
(Iteration 20301 / 49000) loss: 2.302585
(Epoch 83 / 200) train acc: 0.101000; val_acc: 0.102000
(Iteration 20401 / 49000) loss: 2.303149
(Iteration 20501 / 49000) loss: 2.302317
(Epoch 84 / 200) train acc: 0.091000; val_acc: 0.078000
(Iteration 20601 / 49000) loss: 2.302169
(Iteration 20701 / 49000) loss: 2.302857
(Iteration 20801 / 49000) loss: 2.302697
(Epoch 85 / 200) train acc: 0.105000; val_acc: 0.078000
```

```
(Iteration 20901 / 49000) loss: 2.302852
(Iteration 21001 / 49000) loss: 2.302283
(Epoch 86 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 21101 / 49000) loss: 2.301513
(Iteration 21201 / 49000) loss: 2.302415
(Iteration 21301 / 49000) loss: 2.301993
(Epoch 87 / 200) train acc: 0.109000; val_acc: 0.078000
(Iteration 21401 / 49000) loss: 2.302096
(Iteration 21501 / 49000) loss: 2.302569
(Epoch 88 / 200) train acc: 0.095000; val_acc: 0.078000
(Iteration 21601 / 49000) loss: 2.302387
(Iteration 21701 / 49000) loss: 2.302667
(Iteration 21801 / 49000) loss: 2.302487
(Epoch 89 / 200) train acc: 0.092000; val_acc: 0.078000
(Iteration 21901 / 49000) loss: 2.302610
(Iteration 22001 / 49000) loss: 2.302122
(Epoch 90 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 22101 / 49000) loss: 2.302069
(Iteration 22201 / 49000) loss: 2.302245
(Epoch 91 / 200) train acc: 0.101000; val_acc: 0.102000
(Iteration 22301 / 49000) loss: 2.302455
(Iteration 22401 / 49000) loss: 2.302652
(Iteration 22501 / 49000) loss: 2.301879
(Epoch 92 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 22601 / 49000) loss: 2.301922
(Iteration 22701 / 49000) loss: 2.302326
(Epoch 93 / 200) train acc: 0.087000; val_acc: 0.102000
(Iteration 22801 / 49000) loss: 2.302858
(Iteration 22901 / 49000) loss: 2.301854
(Iteration 23001 / 49000) loss: 2.302262
(Epoch 94 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 23101 / 49000) loss: 2.302384
(Iteration 23201 / 49000) loss: 2.302084
(Epoch 95 / 200) train acc: 0.120000; val_acc: 0.078000
(Iteration 23301 / 49000) loss: 2.302558
(Iteration 23401 / 49000) loss: 2.302445
(Iteration 23501 / 49000) loss: 2.302962
(Epoch 96 / 200) train acc: 0.088000; val_acc: 0.078000
(Iteration 23601 / 49000) loss: 2.302586
(Iteration 23701 / 49000) loss: 2.302207
(Epoch 97 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 23801 / 49000) loss: 2.302114
(Iteration 23901 / 49000) loss: 2.302807
(Iteration 24001 / 49000) loss: 2.302710
(Epoch 98 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 24101 / 49000) loss: 2.302772
(Iteration 24201 / 49000) loss: 2.302751
(Epoch 99 / 200) train acc: 0.113000; val_acc: 0.078000
```

```
(Iteration 24301 / 49000) loss: 2.302670
(Iteration 24401 / 49000) loss: 2.302647
(Epoch 100 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 24501 / 49000) loss: 2.302780
(Iteration 24601 / 49000) loss: 2.302314
(Iteration 24701 / 49000) loss: 2.302616
(Epoch 101 / 200) train acc: 0.089000; val_acc: 0.078000
(Iteration 24801 / 49000) loss: 2.302437
(Iteration 24901 / 49000) loss: 2.302678
(Epoch 102 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 25001 / 49000) loss: 2.303017
(Iteration 25101 / 49000) loss: 2.302782
(Iteration 25201 / 49000) loss: 2.302869
(Epoch 103 / 200) train acc: 0.083000; val_acc: 0.078000
(Iteration 25301 / 49000) loss: 2.302783
(Iteration 25401 / 49000) loss: 2.302591
(Epoch 104 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 25501 / 49000) loss: 2.302780
(Iteration 25601 / 49000) loss: 2.302654
(Iteration 25701 / 49000) loss: 2.302969
(Epoch 105 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 25801 / 49000) loss: 2.302518
(Iteration 25901 / 49000) loss: 2.302784
(Epoch 106 / 200) train acc: 0.086000; val_acc: 0.078000
(Iteration 26001 / 49000) loss: 2.302394
(Iteration 26101 / 49000) loss: 2.302689
(Iteration 26201 / 49000) loss: 2.302425
(Epoch 107 / 200) train acc: 0.118000; val_acc: 0.078000
(Iteration 26301 / 49000) loss: 2.302396
(Iteration 26401 / 49000) loss: 2.302507
(Epoch 108 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 26501 / 49000) loss: 2.302507
(Iteration 26601 / 49000) loss: 2.302762
(Iteration 26701 / 49000) loss: 2.302724
(Epoch 109 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 26801 / 49000) loss: 2.302422
(Iteration 26901 / 49000) loss: 2.302659
(Epoch 110 / 200) train acc: 0.112000; val_acc: 0.078000
(Iteration 27001 / 49000) loss: 2.302228
(Iteration 27101 / 49000) loss: 2.302987
(Epoch 111 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 27201 / 49000) loss: 2.302592
(Iteration 27301 / 49000) loss: 2.302261
(Iteration 27401 / 49000) loss: 2.302705
(Epoch 112 / 200) train acc: 0.099000; val_acc: 0.078000
(Iteration 27501 / 49000) loss: 2.302642
(Iteration 27601 / 49000) loss: 2.302816
(Epoch 113 / 200) train acc: 0.112000; val_acc: 0.078000
```

```
(Iteration 27701 / 49000) loss: 2.302747
(Iteration 27801 / 49000) loss: 2.302399
(Iteration 27901 / 49000) loss: 2.302372
(Epoch 114 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 28001 / 49000) loss: 2.302705
(Iteration 28101 / 49000) loss: 2.302703
(Epoch 115 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 28201 / 49000) loss: 2.302380
(Iteration 28301 / 49000) loss: 2.302156
(Iteration 28401 / 49000) loss: 2.302829
(Epoch 116 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 28501 / 49000) loss: 2.302448
(Iteration 28601 / 49000) loss: 2.302815
(Epoch 117 / 200) train acc: 0.089000; val_acc: 0.078000
(Iteration 28701 / 49000) loss: 2.302287
(Iteration 28801 / 49000) loss: 2.302782
(Iteration 28901 / 49000) loss: 2.302532
(Epoch 118 / 200) train acc: 0.112000; val_acc: 0.078000
(Iteration 29001 / 49000) loss: 2.302252
(Iteration 29101 / 49000) loss: 2.302382
(Epoch 119 / 200) train acc: 0.092000; val_acc: 0.078000
(Iteration 29201 / 49000) loss: 2.301871
(Iteration 29301 / 49000) loss: 2.302983
(Epoch 120 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 29401 / 49000) loss: 2.302734
(Iteration 29501 / 49000) loss: 2.302633
(Iteration 29601 / 49000) loss: 2.302659
(Epoch 121 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 29701 / 49000) loss: 2.302696
(Iteration 29801 / 49000) loss: 2.302551
(Epoch 122 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 29901 / 49000) loss: 2.302666
(Iteration 30001 / 49000) loss: 2.302939
(Iteration 30101 / 49000) loss: 2.302394
(Epoch 123 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 30201 / 49000) loss: 2.302280
(Iteration 30301 / 49000) loss: 2.302555
(Epoch 124 / 200) train acc: 0.109000; val_acc: 0.078000
(Iteration 30401 / 49000) loss: 2.302391
(Iteration 30501 / 49000) loss: 2.302699
(Iteration 30601 / 49000) loss: 2.302555
(Epoch 125 / 200) train acc: 0.088000; val_acc: 0.102000
(Iteration 30701 / 49000) loss: 2.303080
(Iteration 30801 / 49000) loss: 2.302725
(Epoch 126 / 200) train acc: 0.104000; val_acc: 0.102000
(Iteration 30901 / 49000) loss: 2.302663
(Iteration 31001 / 49000) loss: 2.302797
(Iteration 31101 / 49000) loss: 2.302738
```

```
(Epoch 127 / 200) train acc: 0.083000; val_acc: 0.105000
(Iteration 31201 / 49000) loss: 2.302985
(Iteration 31301 / 49000) loss: 2.302673
(Epoch 128 / 200) train acc: 0.124000; val_acc: 0.078000
(Iteration 31401 / 49000) loss: 2.302317
(Iteration 31501 / 49000) loss: 2.302612
(Iteration 31601 / 49000) loss: 2.302379
(Epoch 129 / 200) train acc: 0.099000; val_acc: 0.078000
(Iteration 31701 / 49000) loss: 2.302341
(Iteration 31801 / 49000) loss: 2.302827
(Epoch 130 / 200) train acc: 0.119000; val_acc: 0.087000
(Iteration 31901 / 49000) loss: 2.302387
(Iteration 32001 / 49000) loss: 2.302604
(Epoch 131 / 200) train acc: 0.115000; val_acc: 0.078000
(Iteration 32101 / 49000) loss: 2.302635
(Iteration 32201 / 49000) loss: 2.302973
(Iteration 32301 / 49000) loss: 2.302535
(Epoch 132 / 200) train acc: 0.092000; val_acc: 0.078000
(Iteration 32401 / 49000) loss: 2.302772
(Iteration 32501 / 49000) loss: 2.302530
(Epoch 133 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 32601 / 49000) loss: 2.302902
(Iteration 32701 / 49000) loss: 2.302644
(Iteration 32801 / 49000) loss: 2.302250
(Epoch 134 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 32901 / 49000) loss: 2.302934
(Iteration 33001 / 49000) loss: 2.302701
(Epoch 135 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 33101 / 49000) loss: 2.302087
(Iteration 33201 / 49000) loss: 2.302537
(Iteration 33301 / 49000) loss: 2.302075
(Epoch 136 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 33401 / 49000) loss: 2.302486
(Iteration 33501 / 49000) loss: 2.302612
(Epoch 137 / 200) train acc: 0.109000; val_acc: 0.078000
(Iteration 33601 / 49000) loss: 2.302742
(Iteration 33701 / 49000) loss: 2.302846
(Iteration 33801 / 49000) loss: 2.302300
(Epoch 138 / 200) train acc: 0.109000; val_acc: 0.078000
(Iteration 33901 / 49000) loss: 2.302259
(Iteration 34001 / 49000) loss: 2.302360
(Epoch 139 / 200) train acc: 0.082000; val_acc: 0.105000
(Iteration 34101 / 49000) loss: 2.302252
(Iteration 34201 / 49000) loss: 2.302639
(Epoch 140 / 200) train acc: 0.095000; val_acc: 0.105000
(Iteration 34301 / 49000) loss: 2.302652
(Iteration 34401 / 49000) loss: 2.302581
(Iteration 34501 / 49000) loss: 2.302685
```

```
(Epoch 141 / 200) train acc: 0.100000; val_acc: 0.105000
(Iteration 34601 / 49000) loss: 2.302554
(Iteration 34701 / 49000) loss: 2.302590
(Epoch 142 / 200) train acc: 0.091000; val_acc: 0.105000
(Iteration 34801 / 49000) loss: 2.302619
(Iteration 34901 / 49000) loss: 2.302578
(Iteration 35001 / 49000) loss: 2.302585
(Epoch 143 / 200) train acc: 0.106000; val_acc: 0.105000
(Iteration 35101 / 49000) loss: 2.302542
(Iteration 35201 / 49000) loss: 2.302747
(Epoch 144 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 35301 / 49000) loss: 2.303175
(Iteration 35401 / 49000) loss: 2.302502
(Iteration 35501 / 49000) loss: 2.302349
(Epoch 145 / 200) train acc: 0.083000; val_acc: 0.078000
(Iteration 35601 / 49000) loss: 2.302523
(Iteration 35701 / 49000) loss: 2.302767
(Epoch 146 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 35801 / 49000) loss: 2.302459
(Iteration 35901 / 49000) loss: 2.302599
(Iteration 36001 / 49000) loss: 2.302283
(Epoch 147 / 200) train acc: 0.091000; val_acc: 0.087000
(Iteration 36101 / 49000) loss: 2.302764
(Iteration 36201 / 49000) loss: 2.302590
(Epoch 148 / 200) train acc: 0.110000; val_acc: 0.087000
(Iteration 36301 / 49000) loss: 2.302466
(Iteration 36401 / 49000) loss: 2.303036
(Iteration 36501 / 49000) loss: 2.302460
(Epoch 149 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 36601 / 49000) loss: 2.302432
(Iteration 36701 / 49000) loss: 2.302503
(Epoch 150 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 36801 / 49000) loss: 2.302657
(Iteration 36901 / 49000) loss: 2.302766
(Epoch 151 / 200) train acc: 0.114000; val_acc: 0.087000
(Iteration 37001 / 49000) loss: 2.302526
(Iteration 37101 / 49000) loss: 2.302353
(Iteration 37201 / 49000) loss: 2.302602
(Epoch 152 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 37301 / 49000) loss: 2.303046
(Iteration 37401 / 49000) loss: 2.302590
(Epoch 153 / 200) train acc: 0.099000; val_acc: 0.087000
(Iteration 37501 / 49000) loss: 2.302362
(Iteration 37601 / 49000) loss: 2.303086
(Iteration 37701 / 49000) loss: 2.302747
(Epoch 154 / 200) train acc: 0.106000; val_acc: 0.087000
(Iteration 37801 / 49000) loss: 2.302579
(Iteration 37901 / 49000) loss: 2.302873
```

```
(Epoch 155 / 200) train acc: 0.088000; val_acc: 0.087000
(Iteration 38001 / 49000) loss: 2.302785
(Iteration 38101 / 49000) loss: 2.302758
(Iteration 38201 / 49000) loss: 2.302190
(Epoch 156 / 200) train acc: 0.120000; val_acc: 0.087000
(Iteration 38301 / 49000) loss: 2.302644
(Iteration 38401 / 49000) loss: 2.302249
(Epoch 157 / 200) train acc: 0.093000; val_acc: 0.087000
(Iteration 38501 / 49000) loss: 2.302244
(Iteration 38601 / 49000) loss: 2.302016
(Iteration 38701 / 49000) loss: 2.302348
(Epoch 158 / 200) train acc: 0.119000; val_acc: 0.087000
(Iteration 38801 / 49000) loss: 2.302464
(Iteration 38901 / 49000) loss: 2.302838
(Epoch 159 / 200) train acc: 0.084000; val_acc: 0.087000
(Iteration 39001 / 49000) loss: 2.302334
(Iteration 39101 / 49000) loss: 2.302821
(Epoch 160 / 200) train acc: 0.110000; val_acc: 0.087000
(Iteration 39201 / 49000) loss: 2.303092
(Iteration 39301 / 49000) loss: 2.302695
(Iteration 39401 / 49000) loss: 2.302589
(Epoch 161 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 39501 / 49000) loss: 2.302970
(Iteration 39601 / 49000) loss: 2.302610
(Epoch 162 / 200) train acc: 0.091000; val_acc: 0.102000
(Iteration 39701 / 49000) loss: 2.302721
(Iteration 39801 / 49000) loss: 2.302908
(Iteration 39901 / 49000) loss: 2.302560
(Epoch 163 / 200) train acc: 0.100000; val_acc: 0.102000
(Iteration 40001 / 49000) loss: 2.302759
(Iteration 40101 / 49000) loss: 2.302465
(Epoch 164 / 200) train acc: 0.101000; val_acc: 0.102000
(Iteration 40201 / 49000) loss: 2.302432
(Iteration 40301 / 49000) loss: 2.302800
(Iteration 40401 / 49000) loss: 2.302953
(Epoch 165 / 200) train acc: 0.102000; val_acc: 0.102000
(Iteration 40501 / 49000) loss: 2.303001
(Iteration 40601 / 49000) loss: 2.302318
(Epoch 166 / 200) train acc: 0.110000; val_acc: 0.102000
(Iteration 40701 / 49000) loss: 2.302747
(Iteration 40801 / 49000) loss: 2.302849
(Iteration 40901 / 49000) loss: 2.302291
(Epoch 167 / 200) train acc: 0.086000; val_acc: 0.102000
(Iteration 41001 / 49000) loss: 2.302472
(Iteration 41101 / 49000) loss: 2.302217
(Epoch 168 / 200) train acc: 0.104000; val_acc: 0.102000
(Iteration 41201 / 49000) loss: 2.302442
(Iteration 41301 / 49000) loss: 2.302650
```

```
(Iteration 41401 / 49000) loss: 2.302604
(Epoch 169 / 200) train acc: 0.105000; val_acc: 0.098000
(Iteration 41501 / 49000) loss: 2.302770
(Iteration 41601 / 49000) loss: 2.302259
(Epoch 170 / 200) train acc: 0.103000; val_acc: 0.102000
(Iteration 41701 / 49000) loss: 2.302434
(Iteration 41801 / 49000) loss: 2.302986
(Epoch 171 / 200) train acc: 0.113000; val_acc: 0.098000
(Iteration 41901 / 49000) loss: 2.302723
(Iteration 42001 / 49000) loss: 2.302621
(Iteration 42101 / 49000) loss: 2.302615
(Epoch 172 / 200) train acc: 0.108000; val_acc: 0.098000
(Iteration 42201 / 49000) loss: 2.302752
(Iteration 42301 / 49000) loss: 2.302515
(Epoch 173 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 42401 / 49000) loss: 2.302401
(Iteration 42501 / 49000) loss: 2.302224
(Iteration 42601 / 49000) loss: 2.302484
(Epoch 174 / 200) train acc: 0.100000; val_acc: 0.098000
(Iteration 42701 / 49000) loss: 2.302872
(Iteration 42801 / 49000) loss: 2.302773
(Epoch 175 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 42901 / 49000) loss: 2.302516
(Iteration 43001 / 49000) loss: 2.302355
(Iteration 43101 / 49000) loss: 2.302622
(Epoch 176 / 200) train acc: 0.089000; val_acc: 0.098000
(Iteration 43201 / 49000) loss: 2.302656
(Iteration 43301 / 49000) loss: 2.302471
(Epoch 177 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 43401 / 49000) loss: 2.302457
(Iteration 43501 / 49000) loss: 2.302590
(Iteration 43601 / 49000) loss: 2.302578
(Epoch 178 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 43701 / 49000) loss: 2.302938
(Iteration 43801 / 49000) loss: 2.302462
(Epoch 179 / 200) train acc: 0.126000; val_acc: 0.078000
(Iteration 43901 / 49000) loss: 2.303058
(Iteration 44001 / 49000) loss: 2.302596
(Epoch 180 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 44101 / 49000) loss: 2.302335
(Iteration 44201 / 49000) loss: 2.302540
(Iteration 44301 / 49000) loss: 2.302487
(Epoch 181 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 44401 / 49000) loss: 2.302388
(Iteration 44501 / 49000) loss: 2.302622
(Epoch 182 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 44601 / 49000) loss: 2.302754
(Iteration 44701 / 49000) loss: 2.302622
```

```
(Iteration 44801 / 49000) loss: 2.302919
(Epoch 183 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 44901 / 49000) loss: 2.302219
(Iteration 45001 / 49000) loss: 2.302539
(Epoch 184 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 45101 / 49000) loss: 2.302855
(Iteration 45201 / 49000) loss: 2.302704
(Iteration 45301 / 49000) loss: 2.302554
(Epoch 185 / 200) train acc: 0.095000; val_acc: 0.078000
(Iteration 45401 / 49000) loss: 2.302607
(Iteration 45501 / 49000) loss: 2.302504
(Epoch 186 / 200) train acc: 0.101000; val_acc: 0.098000
(Iteration 45601 / 49000) loss: 2.302488
(Iteration 45701 / 49000) loss: 2.302963
(Iteration 45801 / 49000) loss: 2.302249
(Epoch 187 / 200) train acc: 0.133000; val_acc: 0.098000
(Iteration 45901 / 49000) loss: 2.302384
(Iteration 46001 / 49000) loss: 2.302698
(Epoch 188 / 200) train acc: 0.107000; val_acc: 0.098000
(Iteration 46101 / 49000) loss: 2.302619
(Iteration 46201 / 49000) loss: 2.302729
(Iteration 46301 / 49000) loss: 2.302874
(Epoch 189 / 200) train acc: 0.100000; val_acc: 0.078000
(Iteration 46401 / 49000) loss: 2.302686
(Iteration 46501 / 49000) loss: 2.302568
(Epoch 190 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 46601 / 49000) loss: 2.302848
(Iteration 46701 / 49000) loss: 2.302945
(Epoch 191 / 200) train acc: 0.109000; val_acc: 0.098000
(Iteration 46801 / 49000) loss: 2.302860
(Iteration 46901 / 49000) loss: 2.302570
(Iteration 47001 / 49000) loss: 2.302703
(Epoch 192 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 47101 / 49000) loss: 2.302572
(Iteration 47201 / 49000) loss: 2.302621
(Epoch 193 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 47301 / 49000) loss: 2.302625
(Iteration 47401 / 49000) loss: 2.302439
(Iteration 47501 / 49000) loss: 2.302570
(Epoch 194 / 200) train acc: 0.085000; val_acc: 0.078000
(Iteration 47601 / 49000) loss: 2.302767
(Iteration 47701 / 49000) loss: 2.302615
(Epoch 195 / 200) train acc: 0.115000; val_acc: 0.098000
(Iteration 47801 / 49000) loss: 2.302455
(Iteration 47901 / 49000) loss: 2.302381
(Iteration 48001 / 49000) loss: 2.302368
(Epoch 196 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 48101 / 49000) loss: 2.302495
```

```
(Iteration 48201 / 49000) loss: 2.302823
(Epoch 197 / 200) train acc: 0.117000; val_acc: 0.078000
(Iteration 48301 / 49000) loss: 2.302569
(Iteration 48401 / 49000) loss: 2.302653
(Iteration 48501 / 49000) loss: 2.302662
(Epoch 198 / 200) train acc: 0.103000; val_acc: 0.098000
(Iteration 48601 / 49000) loss: 2.302544
(Iteration 48701 / 49000) loss: 2.302702
(Epoch 199 / 200) train acc: 0.085000; val_acc: 0.098000
(Iteration 48801 / 49000) loss: 2.302853
(Iteration 48901 / 49000) loss: 2.302701
(Epoch 200 / 200) train acc: 0.101000; val_acc: 0.098000
(Iteration 1 / 49000) loss: 530.956065
(Epoch 0 / 200) train acc: 0.127000; val_acc: 0.124000
(Iteration 101 / 49000) loss: 74.039420
(Iteration 201 / 49000) loss: 12.149432
(Epoch 1 / 200) train acc: 0.403000; val_acc: 0.396000
(Iteration 301 / 49000) loss: 3.699942
(Iteration 401 / 49000) loss: 2.507810
(Epoch 2 / 200) train acc: 0.137000; val_acc: 0.147000
(Iteration 501 / 49000) loss: 2.333316
(Iteration 601 / 49000) loss: 2.307575
(Iteration 701 / 49000) loss: 2.304335
(Epoch 3 / 200) train acc: 0.096000; val_acc: 0.098000
(Iteration 801 / 49000) loss: 2.302767
(Iteration 901 / 49000) loss: 2.302355
(Epoch 4 / 200) train acc: 0.090000; val_acc: 0.102000
(Iteration 1001 / 49000) loss: 2.303110
(Iteration 1101 / 49000) loss: 2.303304
(Iteration 1201 / 49000) loss: 2.302735
(Epoch 5 / 200) train acc: 0.097000; val_acc: 0.102000
(Iteration 1301 / 49000) loss: 2.302900
(Iteration 1401 / 49000) loss: 2.302583
(Epoch 6 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 1501 / 49000) loss: 2.302945
(Iteration 1601 / 49000) loss: 2.302894
(Iteration 1701 / 49000) loss: 2.302049
(Epoch 7 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 1801 / 49000) loss: 2.302323
(Iteration 1901 / 49000) loss: 2.302537
(Epoch 8 / 200) train acc: 0.096000; val_acc: 0.102000
(Iteration 2001 / 49000) loss: 2.302700
(Iteration 2101 / 49000) loss: 2.302133
(Iteration 2201 / 49000) loss: 2.302703
(Epoch 9 / 200) train acc: 0.095000; val_acc: 0.078000
(Iteration 2301 / 49000) loss: 2.302202
(Iteration 2401 / 49000) loss: 2.302719
(Epoch 10 / 200) train acc: 0.096000; val_acc: 0.078000
```

```
(Iteration 2501 / 49000) loss: 2.302808
(Iteration 2601 / 49000) loss: 2.302580
(Epoch 11 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 2701 / 49000) loss: 2.302576
(Iteration 2801 / 49000) loss: 2.302950
(Iteration 2901 / 49000) loss: 2.302667
(Epoch 12 / 200) train acc: 0.083000; val_acc: 0.078000
(Iteration 3001 / 49000) loss: 2.302835
(Iteration 3101 / 49000) loss: 2.302643
(Epoch 13 / 200) train acc: 0.098000; val_acc: 0.107000
(Iteration 3201 / 49000) loss: 2.302208
(Iteration 3301 / 49000) loss: 2.302156
(Iteration 3401 / 49000) loss: 2.302444
(Epoch 14 / 200) train acc: 0.105000; val_acc: 0.087000
(Iteration 3501 / 49000) loss: 2.302556
(Iteration 3601 / 49000) loss: 2.302942
(Epoch 15 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 3701 / 49000) loss: 2.302629
(Iteration 3801 / 49000) loss: 2.302478
(Iteration 3901 / 49000) loss: 2.302528
(Epoch 16 / 200) train acc: 0.106000; val_acc: 0.087000
(Iteration 4001 / 49000) loss: 2.302851
(Iteration 4101 / 49000) loss: 2.302562
(Epoch 17 / 200) train acc: 0.104000; val_acc: 0.112000
(Iteration 4201 / 49000) loss: 2.302464
(Iteration 4301 / 49000) loss: 2.302861
(Iteration 4401 / 49000) loss: 2.302672
(Epoch 18 / 200) train acc: 0.094000; val_acc: 0.087000
(Iteration 4501 / 49000) loss: 2.302704
(Iteration 4601 / 49000) loss: 2.302516
(Epoch 19 / 200) train acc: 0.110000; val_acc: 0.087000
(Iteration 4701 / 49000) loss: 2.302471
(Iteration 4801 / 49000) loss: 2.302475
(Epoch 20 / 200) train acc: 0.099000; val_acc: 0.087000
(Iteration 4901 / 49000) loss: 2.302356
(Iteration 5001 / 49000) loss: 2.302421
(Iteration 5101 / 49000) loss: 2.303051
(Epoch 21 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 5201 / 49000) loss: 2.302923
(Iteration 5301 / 49000) loss: 2.302648
(Epoch 22 / 200) train acc: 0.109000; val_acc: 0.087000
(Iteration 5401 / 49000) loss: 2.302598
(Iteration 5501 / 49000) loss: 2.302459
(Iteration 5601 / 49000) loss: 2.302600
(Epoch 23 / 200) train acc: 0.116000; val_acc: 0.087000
(Iteration 5701 / 49000) loss: 2.302592
(Iteration 5801 / 49000) loss: 2.302610
(Epoch 24 / 200) train acc: 0.110000; val_acc: 0.105000
```

```
(Iteration 5901 / 49000) loss: 2.302253
(Iteration 6001 / 49000) loss: 2.302825
(Iteration 6101 / 49000) loss: 2.303066
(Epoch 25 / 200) train acc: 0.111000; val_acc: 0.112000
(Iteration 6201 / 49000) loss: 2.302248
(Iteration 6301 / 49000) loss: 2.302289
(Epoch 26 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 6401 / 49000) loss: 2.302910
(Iteration 6501 / 49000) loss: 2.302970
(Iteration 6601 / 49000) loss: 2.302723
(Epoch 27 / 200) train acc: 0.108000; val_acc: 0.098000
(Iteration 6701 / 49000) loss: 2.302328
(Iteration 6801 / 49000) loss: 2.302844
(Epoch 28 / 200) train acc: 0.084000; val_acc: 0.098000
(Iteration 6901 / 49000) loss: 2.302542
(Iteration 7001 / 49000) loss: 2.302487
(Iteration 7101 / 49000) loss: 2.302344
(Epoch 29 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 7201 / 49000) loss: 2.302395
(Iteration 7301 / 49000) loss: 2.302606
(Epoch 30 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 7401 / 49000) loss: 2.302384
(Iteration 7501 / 49000) loss: 2.302241
(Epoch 31 / 200) train acc: 0.089000; val_acc: 0.079000
(Iteration 7601 / 49000) loss: 2.301796
(Iteration 7701 / 49000) loss: 2.303124
(Iteration 7801 / 49000) loss: 2.303228
(Epoch 32 / 200) train acc: 0.085000; val_acc: 0.079000
(Iteration 7901 / 49000) loss: 2.302438
(Iteration 8001 / 49000) loss: 2.302818
(Epoch 33 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 8101 / 49000) loss: 2.302104
(Iteration 8201 / 49000) loss: 2.301697
(Iteration 8301 / 49000) loss: 2.302291
(Epoch 34 / 200) train acc: 0.115000; val_acc: 0.079000
(Iteration 8401 / 49000) loss: 2.302511
(Iteration 8501 / 49000) loss: 2.302304
(Epoch 35 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 8601 / 49000) loss: 2.302522
(Iteration 8701 / 49000) loss: 2.302414
(Iteration 8801 / 49000) loss: 2.302819
(Epoch 36 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 8901 / 49000) loss: 2.302561
(Iteration 9001 / 49000) loss: 2.302562
(Epoch 37 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 9101 / 49000) loss: 2.302756
(Iteration 9201 / 49000) loss: 2.302358
(Iteration 9301 / 49000) loss: 2.302455
```

```
(Epoch 38 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 9401 / 49000) loss: 2.302655
(Iteration 9501 / 49000) loss: 2.302910
(Epoch 39 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 9601 / 49000) loss: 2.302687
(Iteration 9701 / 49000) loss: 2.302498
(Epoch 40 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 9801 / 49000) loss: 2.302604
(Iteration 9901 / 49000) loss: 2.302537
(Iteration 10001 / 49000) loss: 2.302624
(Epoch 41 / 200) train acc: 0.078000; val_acc: 0.079000
(Iteration 10101 / 49000) loss: 2.302377
(Iteration 10201 / 49000) loss: 2.302492
(Epoch 42 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 10301 / 49000) loss: 2.302205
(Iteration 10401 / 49000) loss: 2.302533
(Iteration 10501 / 49000) loss: 2.302699
(Epoch 43 / 200) train acc: 0.081000; val_acc: 0.079000
(Iteration 10601 / 49000) loss: 2.302590
(Iteration 10701 / 49000) loss: 2.302002
(Epoch 44 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 10801 / 49000) loss: 2.302542
(Iteration 10901 / 49000) loss: 2.302437
(Iteration 11001 / 49000) loss: 2.302785
(Epoch 45 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 11101 / 49000) loss: 2.302411
(Iteration 11201 / 49000) loss: 2.302609
(Epoch 46 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 11301 / 49000) loss: 2.302602
(Iteration 11401 / 49000) loss: 2.302914
(Iteration 11501 / 49000) loss: 2.302264
(Epoch 47 / 200) train acc: 0.086000; val_acc: 0.079000
(Iteration 11601 / 49000) loss: 2.302154
(Iteration 11701 / 49000) loss: 2.302787
(Epoch 48 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 11801 / 49000) loss: 2.302697
(Iteration 11901 / 49000) loss: 2.302660
(Iteration 12001 / 49000) loss: 2.302306
(Epoch 49 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 12101 / 49000) loss: 2.302401
(Iteration 12201 / 49000) loss: 2.302418
(Epoch 50 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 12301 / 49000) loss: 2.303057
(Iteration 12401 / 49000) loss: 2.302970
(Epoch 51 / 200) train acc: 0.116000; val_acc: 0.079000
(Iteration 12501 / 49000) loss: 2.302485
(Iteration 12601 / 49000) loss: 2.302429
(Iteration 12701 / 49000) loss: 2.302704
```

```
(Epoch 52 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 12801 / 49000) loss: 2.302706
(Iteration 12901 / 49000) loss: 2.302738
(Epoch 53 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 13001 / 49000) loss: 2.302338
(Iteration 13101 / 49000) loss: 2.302546
(Iteration 13201 / 49000) loss: 2.302616
(Epoch 54 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 13301 / 49000) loss: 2.302490
(Iteration 13401 / 49000) loss: 2.302911
(Epoch 55 / 200) train acc: 0.082000; val_acc: 0.079000
(Iteration 13501 / 49000) loss: 2.302545
(Iteration 13601 / 49000) loss: 2.302813
(Iteration 13701 / 49000) loss: 2.302409
(Epoch 56 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 13801 / 49000) loss: 2.302206
(Iteration 13901 / 49000) loss: 2.302833
(Epoch 57 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 14001 / 49000) loss: 2.302830
(Iteration 14101 / 49000) loss: 2.302640
(Iteration 14201 / 49000) loss: 2.302414
(Epoch 58 / 200) train acc: 0.084000; val_acc: 0.078000
(Iteration 14301 / 49000) loss: 2.302704
(Iteration 14401 / 49000) loss: 2.302633
(Epoch 59 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 14501 / 49000) loss: 2.302354
(Iteration 14601 / 49000) loss: 2.302227
(Epoch 60 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 14701 / 49000) loss: 2.302459
(Iteration 14801 / 49000) loss: 2.302356
(Iteration 14901 / 49000) loss: 2.302987
(Epoch 61 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 15001 / 49000) loss: 2.302408
(Iteration 15101 / 49000) loss: 2.302581
(Epoch 62 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 15201 / 49000) loss: 2.302756
(Iteration 15301 / 49000) loss: 2.302747
(Iteration 15401 / 49000) loss: 2.302485
(Epoch 63 / 200) train acc: 0.088000; val_acc: 0.078000
(Iteration 15501 / 49000) loss: 2.302606
(Iteration 15601 / 49000) loss: 2.303036
(Epoch 64 / 200) train acc: 0.100000; val_acc: 0.078000
(Iteration 15701 / 49000) loss: 2.302131
(Iteration 15801 / 49000) loss: 2.302930
(Iteration 15901 / 49000) loss: 2.303522
(Epoch 65 / 200) train acc: 0.086000; val_acc: 0.078000
(Iteration 16001 / 49000) loss: 2.302361
(Iteration 16101 / 49000) loss: 2.302449
```

```
(Epoch 66 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 16201 / 49000) loss: 2.302836
(Iteration 16301 / 49000) loss: 2.302522
(Iteration 16401 / 49000) loss: 2.302669
(Epoch 67 / 200) train acc: 0.119000; val_acc: 0.078000
(Iteration 16501 / 49000) loss: 2.302645
(Iteration 16601 / 49000) loss: 2.302855
(Epoch 68 / 200) train acc: 0.091000; val_acc: 0.078000
(Iteration 16701 / 49000) loss: 2.302224
(Iteration 16801 / 49000) loss: 2.302567
(Iteration 16901 / 49000) loss: 2.302576
(Epoch 69 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 17001 / 49000) loss: 2.302674
(Iteration 17101 / 49000) loss: 2.302615
(Epoch 70 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 17201 / 49000) loss: 2.302597
(Iteration 17301 / 49000) loss: 2.302643
(Epoch 71 / 200) train acc: 0.087000; val_acc: 0.078000
(Iteration 17401 / 49000) loss: 2.302552
(Iteration 17501 / 49000) loss: 2.302603
(Iteration 17601 / 49000) loss: 2.302640
(Epoch 72 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 17701 / 49000) loss: 2.302453
(Iteration 17801 / 49000) loss: 2.302653
(Epoch 73 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 17901 / 49000) loss: 2.302444
(Iteration 18001 / 49000) loss: 2.302877
(Iteration 18101 / 49000) loss: 2.302944
(Epoch 74 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 18201 / 49000) loss: 2.302977
(Iteration 18301 / 49000) loss: 2.302576
(Epoch 75 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 18401 / 49000) loss: 2.302604
(Iteration 18501 / 49000) loss: 2.302796
(Iteration 18601 / 49000) loss: 2.302242
(Epoch 76 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 18701 / 49000) loss: 2.302312
(Iteration 18801 / 49000) loss: 2.302581
(Epoch 77 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 18901 / 49000) loss: 2.302400
(Iteration 19001 / 49000) loss: 2.302695
(Iteration 19101 / 49000) loss: 2.302845
(Epoch 78 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 19201 / 49000) loss: 2.302182
(Iteration 19301 / 49000) loss: 2.302582
(Epoch 79 / 200) train acc: 0.079000; val_acc: 0.087000
(Iteration 19401 / 49000) loss: 2.302928
(Iteration 19501 / 49000) loss: 2.302419
```

```
(Epoch 80 / 200) train acc: 0.111000; val_acc: 0.087000
(Iteration 19601 / 49000) loss: 2.302474
(Iteration 19701 / 49000) loss: 2.302879
(Iteration 19801 / 49000) loss: 2.302735
(Epoch 81 / 200) train acc: 0.126000; val_acc: 0.087000
(Iteration 19901 / 49000) loss: 2.302457
(Iteration 20001 / 49000) loss: 2.302891
(Epoch 82 / 200) train acc: 0.099000; val_acc: 0.087000
(Iteration 20101 / 49000) loss: 2.302390
(Iteration 20201 / 49000) loss: 2.302715
(Iteration 20301 / 49000) loss: 2.302522
(Epoch 83 / 200) train acc: 0.083000; val_acc: 0.087000
(Iteration 20401 / 49000) loss: 2.302818
(Iteration 20501 / 49000) loss: 2.302620
(Epoch 84 / 200) train acc: 0.114000; val_acc: 0.079000
(Iteration 20601 / 49000) loss: 2.302521
(Iteration 20701 / 49000) loss: 2.302729
(Iteration 20801 / 49000) loss: 2.302497
(Epoch 85 / 200) train acc: 0.122000; val_acc: 0.079000
(Iteration 20901 / 49000) loss: 2.302573
(Iteration 21001 / 49000) loss: 2.302844
(Epoch 86 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 21101 / 49000) loss: 2.302765
(Iteration 21201 / 49000) loss: 2.302546
(Iteration 21301 / 49000) loss: 2.302363
(Epoch 87 / 200) train acc: 0.113000; val_acc: 0.079000
(Iteration 21401 / 49000) loss: 2.302412
(Iteration 21501 / 49000) loss: 2.302537
(Epoch 88 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 21601 / 49000) loss: 2.302725
(Iteration 21701 / 49000) loss: 2.302619
(Iteration 21801 / 49000) loss: 2.302786
(Epoch 89 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 21901 / 49000) loss: 2.302616
(Iteration 22001 / 49000) loss: 2.302680
(Epoch 90 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 22101 / 49000) loss: 2.302455
(Iteration 22201 / 49000) loss: 2.302478
(Epoch 91 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 22301 / 49000) loss: 2.302513
(Iteration 22401 / 49000) loss: 2.302846
(Iteration 22501 / 49000) loss: 2.302990
(Epoch 92 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 22601 / 49000) loss: 2.302633
(Iteration 22701 / 49000) loss: 2.302480
(Epoch 93 / 200) train acc: 0.080000; val_acc: 0.079000
(Iteration 22801 / 49000) loss: 2.302882
(Iteration 22901 / 49000) loss: 2.302301
```

```
(Iteration 23001 / 49000) loss: 2.302721
(Epoch 94 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 23101 / 49000) loss: 2.302714
(Iteration 23201 / 49000) loss: 2.302479
(Epoch 95 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 23301 / 49000) loss: 2.302373
(Iteration 23401 / 49000) loss: 2.302931
(Iteration 23501 / 49000) loss: 2.302889
(Epoch 96 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 23601 / 49000) loss: 2.302578
(Iteration 23701 / 49000) loss: 2.302614
(Epoch 97 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 23801 / 49000) loss: 2.302773
(Iteration 23901 / 49000) loss: 2.302386
(Iteration 24001 / 49000) loss: 2.302336
(Epoch 98 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 24101 / 49000) loss: 2.302706
(Iteration 24201 / 49000) loss: 2.302474
(Epoch 99 / 200) train acc: 0.115000; val_acc: 0.079000
(Iteration 24301 / 49000) loss: 2.302764
(Iteration 24401 / 49000) loss: 2.302557
(Epoch 100 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 24501 / 49000) loss: 2.302686
(Iteration 24601 / 49000) loss: 2.302512
(Iteration 24701 / 49000) loss: 2.302893
(Epoch 101 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 24801 / 49000) loss: 2.302414
(Iteration 24901 / 49000) loss: 2.302377
(Epoch 102 / 200) train acc: 0.118000; val_acc: 0.078000
(Iteration 25001 / 49000) loss: 2.302570
(Iteration 25101 / 49000) loss: 2.302556
(Iteration 25201 / 49000) loss: 2.302651
(Epoch 103 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 25301 / 49000) loss: 2.302341
(Iteration 25401 / 49000) loss: 2.302492
(Epoch 104 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 25501 / 49000) loss: 2.302478
(Iteration 25601 / 49000) loss: 2.302245
(Iteration 25701 / 49000) loss: 2.302197
(Epoch 105 / 200) train acc: 0.106000; val_acc: 0.087000
(Iteration 25801 / 49000) loss: 2.303048
(Iteration 25901 / 49000) loss: 2.302844
(Epoch 106 / 200) train acc: 0.091000; val_acc: 0.087000
(Iteration 26001 / 49000) loss: 2.302759
(Iteration 26101 / 49000) loss: 2.302651
(Iteration 26201 / 49000) loss: 2.302351
(Epoch 107 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 26301 / 49000) loss: 2.302371
```

```
(Iteration 26401 / 49000) loss: 2.302623
(Epoch 108 / 200) train acc: 0.090000; val_acc: 0.087000
(Iteration 26501 / 49000) loss: 2.302670
(Iteration 26601 / 49000) loss: 2.302746
(Iteration 26701 / 49000) loss: 2.302727
(Epoch 109 / 200) train acc: 0.106000; val_acc: 0.087000
(Iteration 26801 / 49000) loss: 2.302636
(Iteration 26901 / 49000) loss: 2.302609
(Epoch 110 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 27001 / 49000) loss: 2.302655
(Iteration 27101 / 49000) loss: 2.302639
(Epoch 111 / 200) train acc: 0.091000; val_acc: 0.087000
(Iteration 27201 / 49000) loss: 2.302500
(Iteration 27301 / 49000) loss: 2.302642
(Iteration 27401 / 49000) loss: 2.302718
(Epoch 112 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 27501 / 49000) loss: 2.302583
(Iteration 27601 / 49000) loss: 2.302632
(Epoch 113 / 200) train acc: 0.113000; val_acc: 0.087000
(Iteration 27701 / 49000) loss: 2.302828
(Iteration 27801 / 49000) loss: 2.302451
(Iteration 27901 / 49000) loss: 2.302570
(Epoch 114 / 200) train acc: 0.113000; val_acc: 0.087000
(Iteration 28001 / 49000) loss: 2.302256
(Iteration 28101 / 49000) loss: 2.302553
(Epoch 115 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 28201 / 49000) loss: 2.302727
(Iteration 28301 / 49000) loss: 2.302594
(Iteration 28401 / 49000) loss: 2.302416
(Epoch 116 / 200) train acc: 0.113000; val_acc: 0.087000
(Iteration 28501 / 49000) loss: 2.302708
(Iteration 28601 / 49000) loss: 2.302662
(Epoch 117 / 200) train acc: 0.086000; val_acc: 0.087000
(Iteration 28701 / 49000) loss: 2.302843
(Iteration 28801 / 49000) loss: 2.302829
(Iteration 28901 / 49000) loss: 2.302800
(Epoch 118 / 200) train acc: 0.095000; val_acc: 0.087000
(Iteration 29001 / 49000) loss: 2.302490
(Iteration 29101 / 49000) loss: 2.302983
(Epoch 119 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 29201 / 49000) loss: 2.302554
(Iteration 29301 / 49000) loss: 2.302660
(Epoch 120 / 200) train acc: 0.114000; val_acc: 0.087000
(Iteration 29401 / 49000) loss: 2.302433
(Iteration 29501 / 49000) loss: 2.302604
(Iteration 29601 / 49000) loss: 2.302148
(Epoch 121 / 200) train acc: 0.112000; val_acc: 0.087000
(Iteration 29701 / 49000) loss: 2.302440
```

```
(Iteration 29801 / 49000) loss: 2.302686
(Epoch 122 / 200) train acc: 0.110000; val_acc: 0.087000
(Iteration 29901 / 49000) loss: 2.302517
(Iteration 30001 / 49000) loss: 2.302653
(Iteration 30101 / 49000) loss: 2.302805
(Epoch 123 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 30201 / 49000) loss: 2.302432
(Iteration 30301 / 49000) loss: 2.302394
(Epoch 124 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 30401 / 49000) loss: 2.302396
(Iteration 30501 / 49000) loss: 2.302393
(Iteration 30601 / 49000) loss: 2.302895
(Epoch 125 / 200) train acc: 0.086000; val_acc: 0.087000
(Iteration 30701 / 49000) loss: 2.302783
(Iteration 30801 / 49000) loss: 2.302801
(Epoch 126 / 200) train acc: 0.090000; val_acc: 0.087000
(Iteration 30901 / 49000) loss: 2.302938
(Iteration 31001 / 49000) loss: 2.302395
(Iteration 31101 / 49000) loss: 2.302467
(Epoch 127 / 200) train acc: 0.105000; val_acc: 0.087000
(Iteration 31201 / 49000) loss: 2.302651
(Iteration 31301 / 49000) loss: 2.302171
(Epoch 128 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 31401 / 49000) loss: 2.302641
(Iteration 31501 / 49000) loss: 2.302378
(Iteration 31601 / 49000) loss: 2.302299
(Epoch 129 / 200) train acc: 0.116000; val_acc: 0.087000
(Iteration 31701 / 49000) loss: 2.302677
(Iteration 31801 / 49000) loss: 2.302244
(Epoch 130 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 31901 / 49000) loss: 2.302469
(Iteration 32001 / 49000) loss: 2.302554
(Epoch 131 / 200) train acc: 0.111000; val_acc: 0.087000
(Iteration 32101 / 49000) loss: 2.302657
(Iteration 32201 / 49000) loss: 2.302508
(Iteration 32301 / 49000) loss: 2.302688
(Epoch 132 / 200) train acc: 0.086000; val_acc: 0.087000
(Iteration 32401 / 49000) loss: 2.302562
(Iteration 32501 / 49000) loss: 2.302621
(Epoch 133 / 200) train acc: 0.093000; val_acc: 0.087000
(Iteration 32601 / 49000) loss: 2.302787
(Iteration 32701 / 49000) loss: 2.302881
(Iteration 32801 / 49000) loss: 2.302517
(Epoch 134 / 200) train acc: 0.088000; val_acc: 0.087000
(Iteration 32901 / 49000) loss: 2.302514
(Iteration 33001 / 49000) loss: 2.302598
(Epoch 135 / 200) train acc: 0.084000; val_acc: 0.087000
(Iteration 33101 / 49000) loss: 2.302501
```

```
(Iteration 33201 / 49000) loss: 2.302484
(Iteration 33301 / 49000) loss: 2.302936
(Epoch 136 / 200) train acc: 0.087000; val_acc: 0.087000
(Iteration 33401 / 49000) loss: 2.302626
(Iteration 33501 / 49000) loss: 2.302763
(Epoch 137 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 33601 / 49000) loss: 2.302372
(Iteration 33701 / 49000) loss: 2.302740
(Iteration 33801 / 49000) loss: 2.302601
(Epoch 138 / 200) train acc: 0.093000; val_acc: 0.087000
(Iteration 33901 / 49000) loss: 2.302606
(Iteration 34001 / 49000) loss: 2.302923
(Epoch 139 / 200) train acc: 0.095000; val_acc: 0.078000
(Iteration 34101 / 49000) loss: 2.302767
(Iteration 34201 / 49000) loss: 2.302715
(Epoch 140 / 200) train acc: 0.099000; val_acc: 0.078000
(Iteration 34301 / 49000) loss: 2.302251
(Iteration 34401 / 49000) loss: 2.302520
(Iteration 34501 / 49000) loss: 2.302439
(Epoch 141 / 200) train acc: 0.095000; val_acc: 0.078000
(Iteration 34601 / 49000) loss: 2.302849
(Iteration 34701 / 49000) loss: 2.302633
(Epoch 142 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 34801 / 49000) loss: 2.302671
(Iteration 34901 / 49000) loss: 2.302194
(Iteration 35001 / 49000) loss: 2.302869
(Epoch 143 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 35101 / 49000) loss: 2.302905
(Iteration 35201 / 49000) loss: 2.302559
(Epoch 144 / 200) train acc: 0.087000; val_acc: 0.079000
(Iteration 35301 / 49000) loss: 2.302400
(Iteration 35401 / 49000) loss: 2.302498
(Iteration 35501 / 49000) loss: 2.302265
(Epoch 145 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 35601 / 49000) loss: 2.302431
(Iteration 35701 / 49000) loss: 2.302784
(Epoch 146 / 200) train acc: 0.111000; val_acc: 0.078000
(Iteration 35801 / 49000) loss: 2.302700
(Iteration 35901 / 49000) loss: 2.302433
(Iteration 36001 / 49000) loss: 2.302574
(Epoch 147 / 200) train acc: 0.100000; val_acc: 0.078000
(Iteration 36101 / 49000) loss: 2.302690
(Iteration 36201 / 49000) loss: 2.302454
(Epoch 148 / 200) train acc: 0.117000; val_acc: 0.078000
(Iteration 36301 / 49000) loss: 2.302870
(Iteration 36401 / 49000) loss: 2.302483
(Iteration 36501 / 49000) loss: 2.302274
(Epoch 149 / 200) train acc: 0.109000; val_acc: 0.078000
```

```
(Iteration 36601 / 49000) loss: 2.302627
(Iteration 36701 / 49000) loss: 2.302616
(Epoch 150 / 200) train acc: 0.112000; val_acc: 0.078000
(Iteration 36801 / 49000) loss: 2.302629
(Iteration 36901 / 49000) loss: 2.302282
(Epoch 151 / 200) train acc: 0.099000; val_acc: 0.078000
(Iteration 37001 / 49000) loss: 2.302632
(Iteration 37101 / 49000) loss: 2.302593
(Iteration 37201 / 49000) loss: 2.303052
(Epoch 152 / 200) train acc: 0.091000; val_acc: 0.087000
(Iteration 37301 / 49000) loss: 2.302367
(Iteration 37401 / 49000) loss: 2.302523
(Epoch 153 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 37501 / 49000) loss: 2.302484
(Iteration 37601 / 49000) loss: 2.302250
(Iteration 37701 / 49000) loss: 2.302819
(Epoch 154 / 200) train acc: 0.092000; val_acc: 0.079000
(Iteration 37801 / 49000) loss: 2.302311
(Iteration 37901 / 49000) loss: 2.302479
(Epoch 155 / 200) train acc: 0.087000; val_acc: 0.079000
(Iteration 38001 / 49000) loss: 2.302786
(Iteration 38101 / 49000) loss: 2.302763
(Iteration 38201 / 49000) loss: 2.302211
(Epoch 156 / 200) train acc: 0.110000; val_acc: 0.079000
(Iteration 38301 / 49000) loss: 2.302683
(Iteration 38401 / 49000) loss: 2.302740
(Epoch 157 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 38501 / 49000) loss: 2.302371
(Iteration 38601 / 49000) loss: 2.302744
(Iteration 38701 / 49000) loss: 2.302761
(Epoch 158 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 38801 / 49000) loss: 2.302520
(Iteration 38901 / 49000) loss: 2.302627
(Epoch 159 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 39001 / 49000) loss: 2.302236
(Iteration 39101 / 49000) loss: 2.302658
(Epoch 160 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 39201 / 49000) loss: 2.302594
(Iteration 39301 / 49000) loss: 2.302352
(Iteration 39401 / 49000) loss: 2.302390
(Epoch 161 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 39501 / 49000) loss: 2.302433
(Iteration 39601 / 49000) loss: 2.302613
(Epoch 162 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 39701 / 49000) loss: 2.302840
(Iteration 39801 / 49000) loss: 2.302936
(Iteration 39901 / 49000) loss: 2.302568
(Epoch 163 / 200) train acc: 0.090000; val_acc: 0.087000
```

```
(Iteration 40001 / 49000) loss: 2.302458
(Iteration 40101 / 49000) loss: 2.302422
(Epoch 164 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 40201 / 49000) loss: 2.302456
(Iteration 40301 / 49000) loss: 2.302891
(Iteration 40401 / 49000) loss: 2.302164
(Epoch 165 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 40501 / 49000) loss: 2.302340
(Iteration 40601 / 49000) loss: 2.302450
(Epoch 166 / 200) train acc: 0.106000; val_acc: 0.087000
(Iteration 40701 / 49000) loss: 2.302219
(Iteration 40801 / 49000) loss: 2.302581
(Iteration 40901 / 49000) loss: 2.302699
(Epoch 167 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 41001 / 49000) loss: 2.302892
(Iteration 41101 / 49000) loss: 2.302483
(Epoch 168 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 41201 / 49000) loss: 2.302574
(Iteration 41301 / 49000) loss: 2.302382
(Iteration 41401 / 49000) loss: 2.302827
(Epoch 169 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 41501 / 49000) loss: 2.302713
(Iteration 41601 / 49000) loss: 2.302383
(Epoch 170 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 41701 / 49000) loss: 2.302096
(Iteration 41801 / 49000) loss: 2.302486
(Epoch 171 / 200) train acc: 0.093000; val_acc: 0.087000
(Iteration 41901 / 49000) loss: 2.302716
(Iteration 42001 / 49000) loss: 2.303011
(Iteration 42101 / 49000) loss: 2.302443
(Epoch 172 / 200) train acc: 0.094000; val_acc: 0.087000
(Iteration 42201 / 49000) loss: 2.302731
(Iteration 42301 / 49000) loss: 2.302405
(Epoch 173 / 200) train acc: 0.088000; val_acc: 0.087000
(Iteration 42401 / 49000) loss: 2.302514
(Iteration 42501 / 49000) loss: 2.302226
(Iteration 42601 / 49000) loss: 2.302593
(Epoch 174 / 200) train acc: 0.093000; val_acc: 0.087000
(Iteration 42701 / 49000) loss: 2.302656
(Iteration 42801 / 49000) loss: 2.302539
(Epoch 175 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 42901 / 49000) loss: 2.302483
(Iteration 43001 / 49000) loss: 2.302372
(Iteration 43101 / 49000) loss: 2.302683
(Epoch 176 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 43201 / 49000) loss: 2.302545
(Iteration 43301 / 49000) loss: 2.302696
(Epoch 177 / 200) train acc: 0.103000; val_acc: 0.087000
```

```
(Iteration 43401 / 49000) loss: 2.302667
(Iteration 43501 / 49000) loss: 2.302471
(Iteration 43601 / 49000) loss: 2.302730
(Epoch 178 / 200) train acc: 0.091000; val_acc: 0.087000
(Iteration 43701 / 49000) loss: 2.302128
(Iteration 43801 / 49000) loss: 2.302763
(Epoch 179 / 200) train acc: 0.119000; val_acc: 0.087000
(Iteration 43901 / 49000) loss: 2.302233
(Iteration 44001 / 49000) loss: 2.302878
(Epoch 180 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 44101 / 49000) loss: 2.302627
(Iteration 44201 / 49000) loss: 2.302381
(Iteration 44301 / 49000) loss: 2.302712
(Epoch 181 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 44401 / 49000) loss: 2.302945
(Iteration 44501 / 49000) loss: 2.302907
(Epoch 182 / 200) train acc: 0.090000; val_acc: 0.087000
(Iteration 44601 / 49000) loss: 2.302646
(Iteration 44701 / 49000) loss: 2.302495
(Iteration 44801 / 49000) loss: 2.302548
(Epoch 183 / 200) train acc: 0.108000; val_acc: 0.087000
(Iteration 44901 / 49000) loss: 2.302760
(Iteration 45001 / 49000) loss: 2.302595
(Epoch 184 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 45101 / 49000) loss: 2.302268
(Iteration 45201 / 49000) loss: 2.302635
(Iteration 45301 / 49000) loss: 2.302483
(Epoch 185 / 200) train acc: 0.102000; val_acc: 0.087000
(Iteration 45401 / 49000) loss: 2.302326
(Iteration 45501 / 49000) loss: 2.302650
(Epoch 186 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 45601 / 49000) loss: 2.302282
(Iteration 45701 / 49000) loss: 2.302565
(Iteration 45801 / 49000) loss: 2.302768
(Epoch 187 / 200) train acc: 0.117000; val_acc: 0.087000
(Iteration 45901 / 49000) loss: 2.302616
(Iteration 46001 / 49000) loss: 2.302599
(Epoch 188 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 46101 / 49000) loss: 2.302670
(Iteration 46201 / 49000) loss: 2.302520
(Iteration 46301 / 49000) loss: 2.302523
(Epoch 189 / 200) train acc: 0.116000; val_acc: 0.087000
(Iteration 46401 / 49000) loss: 2.302573
(Iteration 46501 / 49000) loss: 2.302816
(Epoch 190 / 200) train acc: 0.113000; val_acc: 0.087000
(Iteration 46601 / 49000) loss: 2.302628
(Iteration 46701 / 49000) loss: 2.302703
(Epoch 191 / 200) train acc: 0.100000; val_acc: 0.079000
```

```
(Iteration 46801 / 49000) loss: 2.302676
(Iteration 46901 / 49000) loss: 2.302291
(Iteration 47001 / 49000) loss: 2.302469
(Epoch 192 / 200) train acc: 0.091000; val_acc: 0.087000
(Iteration 47101 / 49000) loss: 2.302902
(Iteration 47201 / 49000) loss: 2.302676
(Epoch 193 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 47301 / 49000) loss: 2.302667
(Iteration 47401 / 49000) loss: 2.302534
(Iteration 47501 / 49000) loss: 2.302391
(Epoch 194 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 47601 / 49000) loss: 2.302807
(Iteration 47701 / 49000) loss: 2.302869
(Epoch 195 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 47801 / 49000) loss: 2.302435
(Iteration 47901 / 49000) loss: 2.302655
(Iteration 48001 / 49000) loss: 2.302358
(Epoch 196 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 48101 / 49000) loss: 2.302907
(Iteration 48201 / 49000) loss: 2.302493
(Epoch 197 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 48301 / 49000) loss: 2.302517
(Iteration 48401 / 49000) loss: 2.302661
(Iteration 48501 / 49000) loss: 2.302442
(Epoch 198 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 48601 / 49000) loss: 2.302771
(Iteration 48701 / 49000) loss: 2.302593
(Epoch 199 / 200) train acc: 0.114000; val_acc: 0.079000
(Iteration 48801 / 49000) loss: 2.302619
(Iteration 48901 / 49000) loss: 2.302690
(Epoch 200 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 1 / 49000) loss: 536.080066
(Epoch 0 / 200) train acc: 0.120000; val_acc: 0.113000
(Iteration 101 / 49000) loss: 73.260229
(Iteration 201 / 49000) loss: 11.844910
(Epoch 1 / 200) train acc: 0.389000; val_acc: 0.352000
(Iteration 301 / 49000) loss: 3.630006
(Iteration 401 / 49000) loss: 2.492559
(Epoch 2 / 200) train acc: 0.089000; val_acc: 0.078000
(Iteration 501 / 49000) loss: 2.330758
(Iteration 601 / 49000) loss: 2.306986
(Iteration 701 / 49000) loss: 2.303965
(Epoch 3 / 200) train acc: 0.114000; val_acc: 0.078000
(Iteration 801 / 49000) loss: 2.302338
(Iteration 901 / 49000) loss: 2.302357
(Epoch 4 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 1001 / 49000) loss: 2.301702
(Iteration 1101 / 49000) loss: 2.302832
```

```
(Iteration 1201 / 49000) loss: 2.303002
(Epoch 5 / 200) train acc: 0.091000; val_acc: 0.078000
(Iteration 1301 / 49000) loss: 2.302767
(Iteration 1401 / 49000) loss: 2.302225
(Epoch 6 / 200) train acc: 0.119000; val_acc: 0.078000
(Iteration 1501 / 49000) loss: 2.302178
(Iteration 1601 / 49000) loss: 2.302604
(Iteration 1701 / 49000) loss: 2.302516
(Epoch 7 / 200) train acc: 0.099000; val_acc: 0.112000
(Iteration 1801 / 49000) loss: 2.302440
(Iteration 1901 / 49000) loss: 2.302238
(Epoch 8 / 200) train acc: 0.089000; val_acc: 0.112000
(Iteration 2001 / 49000) loss: 2.302336
(Iteration 2101 / 49000) loss: 2.302708
(Iteration 2201 / 49000) loss: 2.302733
(Epoch 9 / 200) train acc: 0.113000; val_acc: 0.102000
(Iteration 2301 / 49000) loss: 2.302656
(Iteration 2401 / 49000) loss: 2.302057
(Epoch 10 / 200) train acc: 0.116000; val_acc: 0.102000
(Iteration 2501 / 49000) loss: 2.302539
(Iteration 2601 / 49000) loss: 2.302530
(Epoch 11 / 200) train acc: 0.078000; val_acc: 0.102000
(Iteration 2701 / 49000) loss: 2.302736
(Iteration 2801 / 49000) loss: 2.303394
(Iteration 2901 / 49000) loss: 2.302717
(Epoch 12 / 200) train acc: 0.098000; val_acc: 0.102000
(Iteration 3001 / 49000) loss: 2.302825
(Iteration 3101 / 49000) loss: 2.302530
(Epoch 13 / 200) train acc: 0.116000; val_acc: 0.078000
(Iteration 3201 / 49000) loss: 2.302269
(Iteration 3301 / 49000) loss: 2.302893
(Iteration 3401 / 49000) loss: 2.302665
(Epoch 14 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 3501 / 49000) loss: 2.303011
(Iteration 3601 / 49000) loss: 2.302620
(Epoch 15 / 200) train acc: 0.105000; val_acc: 0.087000
(Iteration 3701 / 49000) loss: 2.302517
(Iteration 3801 / 49000) loss: 2.302730
(Iteration 3901 / 49000) loss: 2.302429
(Epoch 16 / 200) train acc: 0.076000; val_acc: 0.087000
(Iteration 4001 / 49000) loss: 2.302785
(Iteration 4101 / 49000) loss: 2.302272
(Epoch 17 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 4201 / 49000) loss: 2.302516
(Iteration 4301 / 49000) loss: 2.302474
(Iteration 4401 / 49000) loss: 2.302384
(Epoch 18 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 4501 / 49000) loss: 2.301610
```

```
(Iteration 4601 / 49000) loss: 2.302981
(Epoch 19 / 200) train acc: 0.102000; val_acc: 0.087000
(Iteration 4701 / 49000) loss: 2.302318
(Iteration 4801 / 49000) loss: 2.303197
(Epoch 20 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 4901 / 49000) loss: 2.302756
(Iteration 5001 / 49000) loss: 2.303064
(Iteration 5101 / 49000) loss: 2.302570
(Epoch 21 / 200) train acc: 0.100000; val_acc: 0.078000
(Iteration 5201 / 49000) loss: 2.302417
(Iteration 5301 / 49000) loss: 2.303061
(Epoch 22 / 200) train acc: 0.084000; val_acc: 0.078000
(Iteration 5401 / 49000) loss: 2.302363
(Iteration 5501 / 49000) loss: 2.302239
(Iteration 5601 / 49000) loss: 2.302902
(Epoch 23 / 200) train acc: 0.092000; val_acc: 0.105000
(Iteration 5701 / 49000) loss: 2.303057
(Iteration 5801 / 49000) loss: 2.302857
(Epoch 24 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 5901 / 49000) loss: 2.302449
(Iteration 6001 / 49000) loss: 2.303192
(Iteration 6101 / 49000) loss: 2.301985
(Epoch 25 / 200) train acc: 0.084000; val_acc: 0.078000
(Iteration 6201 / 49000) loss: 2.302456
(Iteration 6301 / 49000) loss: 2.302374
(Epoch 26 / 200) train acc: 0.100000; val_acc: 0.078000
(Iteration 6401 / 49000) loss: 2.302712
(Iteration 6501 / 49000) loss: 2.302924
(Iteration 6601 / 49000) loss: 2.302822
(Epoch 27 / 200) train acc: 0.115000; val_acc: 0.078000
(Iteration 6701 / 49000) loss: 2.301808
(Iteration 6801 / 49000) loss: 2.302937
(Epoch 28 / 200) train acc: 0.085000; val_acc: 0.078000
(Iteration 6901 / 49000) loss: 2.303097
(Iteration 7001 / 49000) loss: 2.302540
(Iteration 7101 / 49000) loss: 2.302666
(Epoch 29 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 7201 / 49000) loss: 2.301984
(Iteration 7301 / 49000) loss: 2.302361
(Epoch 30 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 7401 / 49000) loss: 2.302390
(Iteration 7501 / 49000) loss: 2.302620
(Epoch 31 / 200) train acc: 0.087000; val_acc: 0.078000
(Iteration 7601 / 49000) loss: 2.302721
(Iteration 7701 / 49000) loss: 2.302635
(Iteration 7801 / 49000) loss: 2.302857
(Epoch 32 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 7901 / 49000) loss: 2.302665
```

```
(Iteration 8001 / 49000) loss: 2.302278
(Epoch 33 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 8101 / 49000) loss: 2.302670
(Iteration 8201 / 49000) loss: 2.303588
(Iteration 8301 / 49000) loss: 2.302066
(Epoch 34 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 8401 / 49000) loss: 2.302478
(Iteration 8501 / 49000) loss: 2.302294
(Epoch 35 / 200) train acc: 0.116000; val_acc: 0.078000
(Iteration 8601 / 49000) loss: 2.303013
(Iteration 8701 / 49000) loss: 2.302760
(Iteration 8801 / 49000) loss: 2.302363
(Epoch 36 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 8901 / 49000) loss: 2.302530
(Iteration 9001 / 49000) loss: 2.302859
(Epoch 37 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 9101 / 49000) loss: 2.302974
(Iteration 9201 / 49000) loss: 2.303121
(Iteration 9301 / 49000) loss: 2.302670
(Epoch 38 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 9401 / 49000) loss: 2.303220
(Iteration 9501 / 49000) loss: 2.302613
(Epoch 39 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 9601 / 49000) loss: 2.302539
(Iteration 9701 / 49000) loss: 2.302436
(Epoch 40 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 9801 / 49000) loss: 2.302622
(Iteration 9901 / 49000) loss: 2.302559
(Iteration 10001 / 49000) loss: 2.302268
(Epoch 41 / 200) train acc: 0.088000; val_acc: 0.078000
(Iteration 10101 / 49000) loss: 2.302223
(Iteration 10201 / 49000) loss: 2.302911
(Epoch 42 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 10301 / 49000) loss: 2.302282
(Iteration 10401 / 49000) loss: 2.302641
(Iteration 10501 / 49000) loss: 2.302661
(Epoch 43 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 10601 / 49000) loss: 2.302784
(Iteration 10701 / 49000) loss: 2.302811
(Epoch 44 / 200) train acc: 0.113000; val_acc: 0.078000
(Iteration 10801 / 49000) loss: 2.303085
(Iteration 10901 / 49000) loss: 2.302264
(Iteration 11001 / 49000) loss: 2.302562
(Epoch 45 / 200) train acc: 0.098000; val_acc: 0.107000
(Iteration 11101 / 49000) loss: 2.302300
(Iteration 11201 / 49000) loss: 2.302722
(Epoch 46 / 200) train acc: 0.104000; val_acc: 0.105000
(Iteration 11301 / 49000) loss: 2.302462
```

```
(Iteration 11401 / 49000) loss: 2.302408
(Iteration 11501 / 49000) loss: 2.302593
(Epoch 47 / 200) train acc: 0.090000; val_acc: 0.105000
(Iteration 11601 / 49000) loss: 2.303018
(Iteration 11701 / 49000) loss: 2.302629
(Epoch 48 / 200) train acc: 0.096000; val_acc: 0.105000
(Iteration 11801 / 49000) loss: 2.302880
(Iteration 11901 / 49000) loss: 2.302489
(Iteration 12001 / 49000) loss: 2.302596
(Epoch 49 / 200) train acc: 0.100000; val_acc: 0.107000
(Iteration 12101 / 49000) loss: 2.302835
(Iteration 12201 / 49000) loss: 2.302202
(Epoch 50 / 200) train acc: 0.105000; val_acc: 0.105000
(Iteration 12301 / 49000) loss: 2.302505
(Iteration 12401 / 49000) loss: 2.302705
(Epoch 51 / 200) train acc: 0.093000; val_acc: 0.112000
(Iteration 12501 / 49000) loss: 2.302827
(Iteration 12601 / 49000) loss: 2.302273
(Iteration 12701 / 49000) loss: 2.301967
(Epoch 52 / 200) train acc: 0.104000; val_acc: 0.112000
(Iteration 12801 / 49000) loss: 2.302479
(Iteration 12901 / 49000) loss: 2.302333
(Epoch 53 / 200) train acc: 0.092000; val_acc: 0.112000
(Iteration 13001 / 49000) loss: 2.302695
(Iteration 13101 / 49000) loss: 2.302634
(Iteration 13201 / 49000) loss: 2.302495
(Epoch 54 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 13301 / 49000) loss: 2.302947
(Iteration 13401 / 49000) loss: 2.302725
(Epoch 55 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 13501 / 49000) loss: 2.302848
(Iteration 13601 / 49000) loss: 2.302661
(Iteration 13701 / 49000) loss: 2.302672
(Epoch 56 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 13801 / 49000) loss: 2.302696
(Iteration 13901 / 49000) loss: 2.302481
(Epoch 57 / 200) train acc: 0.119000; val_acc: 0.079000
(Iteration 14001 / 49000) loss: 2.302700
(Iteration 14101 / 49000) loss: 2.302645
(Iteration 14201 / 49000) loss: 2.302902
(Epoch 58 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 14301 / 49000) loss: 2.302450
(Iteration 14401 / 49000) loss: 2.302609
(Epoch 59 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 14501 / 49000) loss: 2.302651
(Iteration 14601 / 49000) loss: 2.302856
(Epoch 60 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 14701 / 49000) loss: 2.302534
```

```
(Iteration 14801 / 49000) loss: 2.302562
(Iteration 14901 / 49000) loss: 2.302517
(Epoch 61 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 15001 / 49000) loss: 2.302617
(Iteration 15101 / 49000) loss: 2.302486
(Epoch 62 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 15201 / 49000) loss: 2.303047
(Iteration 15301 / 49000) loss: 2.302944
(Iteration 15401 / 49000) loss: 2.302063
(Epoch 63 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 15501 / 49000) loss: 2.302605
(Iteration 15601 / 49000) loss: 2.302250
(Epoch 64 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 15701 / 49000) loss: 2.302559
(Iteration 15801 / 49000) loss: 2.302688
(Iteration 15901 / 49000) loss: 2.302366
(Epoch 65 / 200) train acc: 0.113000; val_acc: 0.079000
(Iteration 16001 / 49000) loss: 2.302853
(Iteration 16101 / 49000) loss: 2.302519
(Epoch 66 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 16201 / 49000) loss: 2.303195
(Iteration 16301 / 49000) loss: 2.302428
(Iteration 16401 / 49000) loss: 2.302795
(Epoch 67 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 16501 / 49000) loss: 2.302559
(Iteration 16601 / 49000) loss: 2.302517
(Epoch 68 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 16701 / 49000) loss: 2.302526
(Iteration 16801 / 49000) loss: 2.302783
(Iteration 16901 / 49000) loss: 2.302442
(Epoch 69 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 17001 / 49000) loss: 2.302415
(Iteration 17101 / 49000) loss: 2.302414
(Epoch 70 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 17201 / 49000) loss: 2.302695
(Iteration 17301 / 49000) loss: 2.302727
(Epoch 71 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 17401 / 49000) loss: 2.302704
(Iteration 17501 / 49000) loss: 2.302739
(Iteration 17601 / 49000) loss: 2.302581
(Epoch 72 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 17701 / 49000) loss: 2.302659
(Iteration 17801 / 49000) loss: 2.302813
(Epoch 73 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 17901 / 49000) loss: 2.302448
(Iteration 18001 / 49000) loss: 2.302619
(Iteration 18101 / 49000) loss: 2.302780
(Epoch 74 / 200) train acc: 0.111000; val_acc: 0.079000
```

```
(Iteration 18201 / 49000) loss: 2.302675
(Iteration 18301 / 49000) loss: 2.302942
(Epoch 75 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 18401 / 49000) loss: 2.302916
(Iteration 18501 / 49000) loss: 2.302841
(Iteration 18601 / 49000) loss: 2.302837
(Epoch 76 / 200) train acc: 0.105000; val_acc: 0.098000
(Iteration 18701 / 49000) loss: 2.302273
(Iteration 18801 / 49000) loss: 2.302365
(Epoch 77 / 200) train acc: 0.103000; val_acc: 0.098000
(Iteration 18901 / 49000) loss: 2.302506
(Iteration 19001 / 49000) loss: 2.302743
(Iteration 19101 / 49000) loss: 2.302082
(Epoch 78 / 200) train acc: 0.092000; val_acc: 0.098000
(Iteration 19201 / 49000) loss: 2.302978
(Iteration 19301 / 49000) loss: 2.302477
(Epoch 79 / 200) train acc: 0.111000; val_acc: 0.079000
(Iteration 19401 / 49000) loss: 2.302654
(Iteration 19501 / 49000) loss: 2.302394
(Epoch 80 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 19601 / 49000) loss: 2.302536
(Iteration 19701 / 49000) loss: 2.302692
(Iteration 19801 / 49000) loss: 2.302677
(Epoch 81 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 19901 / 49000) loss: 2.302306
(Iteration 20001 / 49000) loss: 2.302776
(Epoch 82 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 20101 / 49000) loss: 2.302708
(Iteration 20201 / 49000) loss: 2.302468
(Iteration 20301 / 49000) loss: 2.302723
(Epoch 83 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 20401 / 49000) loss: 2.302739
(Iteration 20501 / 49000) loss: 2.303106
(Epoch 84 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 20601 / 49000) loss: 2.302454
(Iteration 20701 / 49000) loss: 2.302405
(Iteration 20801 / 49000) loss: 2.302832
(Epoch 85 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 20901 / 49000) loss: 2.302651
(Iteration 21001 / 49000) loss: 2.302764
(Epoch 86 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 21101 / 49000) loss: 2.302516
(Iteration 21201 / 49000) loss: 2.302748
(Iteration 21301 / 49000) loss: 2.302317
(Epoch 87 / 200) train acc: 0.081000; val_acc: 0.079000
(Iteration 21401 / 49000) loss: 2.302398
(Iteration 21501 / 49000) loss: 2.303235
(Epoch 88 / 200) train acc: 0.097000; val_acc: 0.079000
```

```
(Iteration 21601 / 49000) loss: 2.302645
(Iteration 21701 / 49000) loss: 2.302389
(Iteration 21801 / 49000) loss: 2.302673
(Epoch 89 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 21901 / 49000) loss: 2.302511
(Iteration 22001 / 49000) loss: 2.302652
(Epoch 90 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 22101 / 49000) loss: 2.302463
(Iteration 22201 / 49000) loss: 2.302434
(Epoch 91 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 22301 / 49000) loss: 2.302909
(Iteration 22401 / 49000) loss: 2.302606
(Iteration 22501 / 49000) loss: 2.302452
(Epoch 92 / 200) train acc: 0.114000; val_acc: 0.079000
(Iteration 22601 / 49000) loss: 2.302761
(Iteration 22701 / 49000) loss: 2.302397
(Epoch 93 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 22801 / 49000) loss: 2.302496
(Iteration 22901 / 49000) loss: 2.302233
(Iteration 23001 / 49000) loss: 2.302417
(Epoch 94 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 23101 / 49000) loss: 2.302566
(Iteration 23201 / 49000) loss: 2.302667
(Epoch 95 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 23301 / 49000) loss: 2.302503
(Iteration 23401 / 49000) loss: 2.302541
(Iteration 23501 / 49000) loss: 2.302790
(Epoch 96 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 23601 / 49000) loss: 2.302706
(Iteration 23701 / 49000) loss: 2.302228
(Epoch 97 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 23801 / 49000) loss: 2.302780
(Iteration 23901 / 49000) loss: 2.302707
(Iteration 24001 / 49000) loss: 2.302523
(Epoch 98 / 200) train acc: 0.096000; val_acc: 0.098000
(Iteration 24101 / 49000) loss: 2.302718
(Iteration 24201 / 49000) loss: 2.302750
(Epoch 99 / 200) train acc: 0.104000; val_acc: 0.098000
(Iteration 24301 / 49000) loss: 2.302518
(Iteration 24401 / 49000) loss: 2.302560
(Epoch 100 / 200) train acc: 0.099000; val_acc: 0.098000
(Iteration 24501 / 49000) loss: 2.302774
(Iteration 24601 / 49000) loss: 2.302909
(Iteration 24701 / 49000) loss: 2.302634
(Epoch 101 / 200) train acc: 0.092000; val_acc: 0.079000
(Iteration 24801 / 49000) loss: 2.302631
(Iteration 24901 / 49000) loss: 2.302583
(Epoch 102 / 200) train acc: 0.095000; val_acc: 0.079000
```

```
(Iteration 25001 / 49000) loss: 2.302465
(Iteration 25101 / 49000) loss: 2.302671
(Iteration 25201 / 49000) loss: 2.302448
(Epoch 103 / 200) train acc: 0.083000; val_acc: 0.079000
(Iteration 25301 / 49000) loss: 2.302244
(Iteration 25401 / 49000) loss: 2.302368
(Epoch 104 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 25501 / 49000) loss: 2.302729
(Iteration 25601 / 49000) loss: 2.302724
(Iteration 25701 / 49000) loss: 2.302465
(Epoch 105 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 25801 / 49000) loss: 2.302489
(Iteration 25901 / 49000) loss: 2.302304
(Epoch 106 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 26001 / 49000) loss: 2.302543
(Iteration 26101 / 49000) loss: 2.302438
(Iteration 26201 / 49000) loss: 2.302839
(Epoch 107 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 26301 / 49000) loss: 2.302703
(Iteration 26401 / 49000) loss: 2.302308
(Epoch 108 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 26501 / 49000) loss: 2.302390
(Iteration 26601 / 49000) loss: 2.302405
(Iteration 26701 / 49000) loss: 2.302458
(Epoch 109 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 26801 / 49000) loss: 2.302668
(Iteration 26901 / 49000) loss: 2.302545
(Epoch 110 / 200) train acc: 0.087000; val_acc: 0.079000
(Iteration 27001 / 49000) loss: 2.302684
(Iteration 27101 / 49000) loss: 2.302395
(Epoch 111 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 27201 / 49000) loss: 2.302445
(Iteration 27301 / 49000) loss: 2.302401
(Iteration 27401 / 49000) loss: 2.302632
(Epoch 112 / 200) train acc: 0.110000; val_acc: 0.087000
(Iteration 27501 / 49000) loss: 2.302507
(Iteration 27601 / 49000) loss: 2.302794
(Epoch 113 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 27701 / 49000) loss: 2.302520
(Iteration 27801 / 49000) loss: 2.302745
(Iteration 27901 / 49000) loss: 2.302525
(Epoch 114 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 28001 / 49000) loss: 2.302775
(Iteration 28101 / 49000) loss: 2.302299
(Epoch 115 / 200) train acc: 0.113000; val_acc: 0.079000
(Iteration 28201 / 49000) loss: 2.302487
(Iteration 28301 / 49000) loss: 2.302345
(Iteration 28401 / 49000) loss: 2.302645
```

```
(Epoch 116 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 28501 / 49000) loss: 2.302912
(Iteration 28601 / 49000) loss: 2.302562
(Epoch 117 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 28701 / 49000) loss: 2.302466
(Iteration 28801 / 49000) loss: 2.302985
(Iteration 28901 / 49000) loss: 2.302665
(Epoch 118 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 29001 / 49000) loss: 2.302723
(Iteration 29101 / 49000) loss: 2.302640
(Epoch 119 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 29201 / 49000) loss: 2.302489
(Iteration 29301 / 49000) loss: 2.302674
(Epoch 120 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 29401 / 49000) loss: 2.302478
(Iteration 29501 / 49000) loss: 2.302563
(Iteration 29601 / 49000) loss: 2.302834
(Epoch 121 / 200) train acc: 0.085000; val_acc: 0.079000
(Iteration 29701 / 49000) loss: 2.302643
(Iteration 29801 / 49000) loss: 2.302707
(Epoch 122 / 200) train acc: 0.092000; val_acc: 0.079000
(Iteration 29901 / 49000) loss: 2.302690
(Iteration 30001 / 49000) loss: 2.302445
(Iteration 30101 / 49000) loss: 2.302915
(Epoch 123 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 30201 / 49000) loss: 2.302694
(Iteration 30301 / 49000) loss: 2.302726
(Epoch 124 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 30401 / 49000) loss: 2.302220
(Iteration 30501 / 49000) loss: 2.302682
(Iteration 30601 / 49000) loss: 2.302613
(Epoch 125 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 30701 / 49000) loss: 2.303085
(Iteration 30801 / 49000) loss: 2.302574
(Epoch 126 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 30901 / 49000) loss: 2.302655
(Iteration 31001 / 49000) loss: 2.302706
(Iteration 31101 / 49000) loss: 2.302496
(Epoch 127 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 31201 / 49000) loss: 2.302789
(Iteration 31301 / 49000) loss: 2.302159
(Epoch 128 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 31401 / 49000) loss: 2.302862
(Iteration 31501 / 49000) loss: 2.302608
(Iteration 31601 / 49000) loss: 2.302759
(Epoch 129 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 31701 / 49000) loss: 2.302901
(Iteration 31801 / 49000) loss: 2.302445
```

```
(Epoch 130 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 31901 / 49000) loss: 2.302713
(Iteration 32001 / 49000) loss: 2.302498
(Epoch 131 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 32101 / 49000) loss: 2.302396
(Iteration 32201 / 49000) loss: 2.302557
(Iteration 32301 / 49000) loss: 2.302703
(Epoch 132 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 32401 / 49000) loss: 2.302741
(Iteration 32501 / 49000) loss: 2.302593
(Epoch 133 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 32601 / 49000) loss: 2.302137
(Iteration 32701 / 49000) loss: 2.302482
(Iteration 32801 / 49000) loss: 2.302526
(Epoch 134 / 200) train acc: 0.110000; val_acc: 0.087000
(Iteration 32901 / 49000) loss: 2.302502
(Iteration 33001 / 49000) loss: 2.303092
(Epoch 135 / 200) train acc: 0.102000; val_acc: 0.087000
(Iteration 33101 / 49000) loss: 2.302711
(Iteration 33201 / 49000) loss: 2.302907
(Iteration 33301 / 49000) loss: 2.302361
(Epoch 136 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 33401 / 49000) loss: 2.302935
(Iteration 33501 / 49000) loss: 2.302432
(Epoch 137 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 33601 / 49000) loss: 2.302321
(Iteration 33701 / 49000) loss: 2.302875
(Iteration 33801 / 49000) loss: 2.302573
(Epoch 138 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 33901 / 49000) loss: 2.302428
(Iteration 34001 / 49000) loss: 2.302595
(Epoch 139 / 200) train acc: 0.111000; val_acc: 0.087000
(Iteration 34101 / 49000) loss: 2.302411
(Iteration 34201 / 49000) loss: 2.302486
(Epoch 140 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 34301 / 49000) loss: 2.302322
(Iteration 34401 / 49000) loss: 2.302778
(Iteration 34501 / 49000) loss: 2.302929
(Epoch 141 / 200) train acc: 0.108000; val_acc: 0.087000
(Iteration 34601 / 49000) loss: 2.302342
(Iteration 34701 / 49000) loss: 2.302543
(Epoch 142 / 200) train acc: 0.093000; val_acc: 0.087000
(Iteration 34801 / 49000) loss: 2.302808
(Iteration 34901 / 49000) loss: 2.302872
(Iteration 35001 / 49000) loss: 2.302330
(Epoch 143 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 35101 / 49000) loss: 2.302406
(Iteration 35201 / 49000) loss: 2.302858
```

```
(Epoch 144 / 200) train acc: 0.112000; val_acc: 0.087000
(Iteration 35301 / 49000) loss: 2.302518
(Iteration 35401 / 49000) loss: 2.302277
(Iteration 35501 / 49000) loss: 2.302564
(Epoch 145 / 200) train acc: 0.083000; val_acc: 0.087000
(Iteration 35601 / 49000) loss: 2.302807
(Iteration 35701 / 49000) loss: 2.302462
(Epoch 146 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 35801 / 49000) loss: 2.302419
(Iteration 35901 / 49000) loss: 2.302046
(Iteration 36001 / 49000) loss: 2.302998
(Epoch 147 / 200) train acc: 0.102000; val_acc: 0.087000
(Iteration 36101 / 49000) loss: 2.302656
(Iteration 36201 / 49000) loss: 2.302780
(Epoch 148 / 200) train acc: 0.110000; val_acc: 0.087000
(Iteration 36301 / 49000) loss: 2.302810
(Iteration 36401 / 49000) loss: 2.302414
(Iteration 36501 / 49000) loss: 2.302832
(Epoch 149 / 200) train acc: 0.106000; val_acc: 0.087000
(Iteration 36601 / 49000) loss: 2.302296
(Iteration 36701 / 49000) loss: 2.302683
(Epoch 150 / 200) train acc: 0.110000; val_acc: 0.087000
(Iteration 36801 / 49000) loss: 2.302659
(Iteration 36901 / 49000) loss: 2.302677
(Epoch 151 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 37001 / 49000) loss: 2.302280
(Iteration 37101 / 49000) loss: 2.303286
(Iteration 37201 / 49000) loss: 2.302389
(Epoch 152 / 200) train acc: 0.116000; val_acc: 0.087000
(Iteration 37301 / 49000) loss: 2.302766
(Iteration 37401 / 49000) loss: 2.302339
(Epoch 153 / 200) train acc: 0.086000; val_acc: 0.087000
(Iteration 37501 / 49000) loss: 2.302885
(Iteration 37601 / 49000) loss: 2.302523
(Iteration 37701 / 49000) loss: 2.302049
(Epoch 154 / 200) train acc: 0.115000; val_acc: 0.087000
(Iteration 37801 / 49000) loss: 2.302667
(Iteration 37901 / 49000) loss: 2.302880
(Epoch 155 / 200) train acc: 0.110000; val_acc: 0.087000
(Iteration 38001 / 49000) loss: 2.302786
(Iteration 38101 / 49000) loss: 2.302566
(Iteration 38201 / 49000) loss: 2.302525
(Epoch 156 / 200) train acc: 0.117000; val_acc: 0.087000
(Iteration 38301 / 49000) loss: 2.302226
(Iteration 38401 / 49000) loss: 2.302773
(Epoch 157 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 38501 / 49000) loss: 2.302443
(Iteration 38601 / 49000) loss: 2.303052
```

```
(Iteration 38701 / 49000) loss: 2.302551
(Epoch 158 / 200) train acc: 0.099000; val_acc: 0.087000
(Iteration 38801 / 49000) loss: 2.302336
(Iteration 38901 / 49000) loss: 2.301959
(Epoch 159 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 39001 / 49000) loss: 2.302549
(Iteration 39101 / 49000) loss: 2.302865
(Epoch 160 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 39201 / 49000) loss: 2.302469
(Iteration 39301 / 49000) loss: 2.302403
(Iteration 39401 / 49000) loss: 2.302288
(Epoch 161 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 39501 / 49000) loss: 2.302638
(Iteration 39601 / 49000) loss: 2.302725
(Epoch 162 / 200) train acc: 0.099000; val_acc: 0.087000
(Iteration 39701 / 49000) loss: 2.301996
(Iteration 39801 / 49000) loss: 2.302796
(Iteration 39901 / 49000) loss: 2.302827
(Epoch 163 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 40001 / 49000) loss: 2.302807
(Iteration 40101 / 49000) loss: 2.302374
(Epoch 164 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 40201 / 49000) loss: 2.302264
(Iteration 40301 / 49000) loss: 2.302419
(Iteration 40401 / 49000) loss: 2.302658
(Epoch 165 / 200) train acc: 0.114000; val_acc: 0.087000
(Iteration 40501 / 49000) loss: 2.302366
(Iteration 40601 / 49000) loss: 2.302379
(Epoch 166 / 200) train acc: 0.085000; val_acc: 0.087000
(Iteration 40701 / 49000) loss: 2.302879
(Iteration 40801 / 49000) loss: 2.302061
(Iteration 40901 / 49000) loss: 2.302426
(Epoch 167 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 41001 / 49000) loss: 2.302529
(Iteration 41101 / 49000) loss: 2.302542
(Epoch 168 / 200) train acc: 0.086000; val_acc: 0.087000
(Iteration 41201 / 49000) loss: 2.302783
(Iteration 41301 / 49000) loss: 2.302567
(Iteration 41401 / 49000) loss: 2.302422
(Epoch 169 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 41501 / 49000) loss: 2.302737
(Iteration 41601 / 49000) loss: 2.302563
(Epoch 170 / 200) train acc: 0.096000; val_acc: 0.087000
(Iteration 41701 / 49000) loss: 2.302987
(Iteration 41801 / 49000) loss: 2.302831
(Epoch 171 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 41901 / 49000) loss: 2.302440
(Iteration 42001 / 49000) loss: 2.302337
```

```
(Iteration 42101 / 49000) loss: 2.302492
(Epoch 172 / 200) train acc: 0.108000; val_acc: 0.087000
(Iteration 42201 / 49000) loss: 2.302313
(Iteration 42301 / 49000) loss: 2.301933
(Epoch 173 / 200) train acc: 0.114000; val_acc: 0.087000
(Iteration 42401 / 49000) loss: 2.302920
(Iteration 42501 / 49000) loss: 2.302623
(Iteration 42601 / 49000) loss: 2.302542
(Epoch 174 / 200) train acc: 0.099000; val_acc: 0.087000
(Iteration 42701 / 49000) loss: 2.302970
(Iteration 42801 / 49000) loss: 2.302680
(Epoch 175 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 42901 / 49000) loss: 2.302408
(Iteration 43001 / 49000) loss: 2.302761
(Iteration 43101 / 49000) loss: 2.302646
(Epoch 176 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 43201 / 49000) loss: 2.302715
(Iteration 43301 / 49000) loss: 2.302922
(Epoch 177 / 200) train acc: 0.086000; val_acc: 0.087000
(Iteration 43401 / 49000) loss: 2.302459
(Iteration 43501 / 49000) loss: 2.302912
(Iteration 43601 / 49000) loss: 2.302189
(Epoch 178 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 43701 / 49000) loss: 2.302364
(Iteration 43801 / 49000) loss: 2.302419
(Epoch 179 / 200) train acc: 0.079000; val_acc: 0.087000
(Iteration 43901 / 49000) loss: 2.302786
(Iteration 44001 / 49000) loss: 2.302679
(Epoch 180 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 44101 / 49000) loss: 2.302408
(Iteration 44201 / 49000) loss: 2.302495
(Iteration 44301 / 49000) loss: 2.302463
(Epoch 181 / 200) train acc: 0.120000; val_acc: 0.087000
(Iteration 44401 / 49000) loss: 2.302337
(Iteration 44501 / 49000) loss: 2.302704
(Epoch 182 / 200) train acc: 0.094000; val_acc: 0.087000
(Iteration 44601 / 49000) loss: 2.302900
(Iteration 44701 / 49000) loss: 2.302640
(Iteration 44801 / 49000) loss: 2.302708
(Epoch 183 / 200) train acc: 0.089000; val_acc: 0.087000
(Iteration 44901 / 49000) loss: 2.302624
(Iteration 45001 / 49000) loss: 2.302946
(Epoch 184 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 45101 / 49000) loss: 2.302481
(Iteration 45201 / 49000) loss: 2.302530
(Iteration 45301 / 49000) loss: 2.302667
(Epoch 185 / 200) train acc: 0.113000; val_acc: 0.087000
(Iteration 45401 / 49000) loss: 2.302709
```

```
(Iteration 45501 / 49000) loss: 2.302381
(Epoch 186 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 45601 / 49000) loss: 2.302681
(Iteration 45701 / 49000) loss: 2.302393
(Iteration 45801 / 49000) loss: 2.302733
(Epoch 187 / 200) train acc: 0.090000; val_acc: 0.087000
(Iteration 45901 / 49000) loss: 2.302411
(Iteration 46001 / 49000) loss: 2.302239
(Epoch 188 / 200) train acc: 0.094000; val_acc: 0.087000
(Iteration 46101 / 49000) loss: 2.302635
(Iteration 46201 / 49000) loss: 2.302682
(Iteration 46301 / 49000) loss: 2.302763
(Epoch 189 / 200) train acc: 0.106000; val_acc: 0.087000
(Iteration 46401 / 49000) loss: 2.302599
(Iteration 46501 / 49000) loss: 2.302442
(Epoch 190 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 46601 / 49000) loss: 2.302339
(Iteration 46701 / 49000) loss: 2.302622
(Epoch 191 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 46801 / 49000) loss: 2.302520
(Iteration 46901 / 49000) loss: 2.302592
(Iteration 47001 / 49000) loss: 2.302172
(Epoch 192 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 47101 / 49000) loss: 2.302636
(Iteration 47201 / 49000) loss: 2.302382
(Epoch 193 / 200) train acc: 0.102000; val_acc: 0.087000
(Iteration 47301 / 49000) loss: 2.302207
(Iteration 47401 / 49000) loss: 2.302497
(Iteration 47501 / 49000) loss: 2.302931
(Epoch 194 / 200) train acc: 0.102000; val_acc: 0.087000
(Iteration 47601 / 49000) loss: 2.302491
(Iteration 47701 / 49000) loss: 2.302697
(Epoch 195 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 47801 / 49000) loss: 2.302762
(Iteration 47901 / 49000) loss: 2.302905
(Iteration 48001 / 49000) loss: 2.302486
(Epoch 196 / 200) train acc: 0.115000; val_acc: 0.087000
(Iteration 48101 / 49000) loss: 2.302717
(Iteration 48201 / 49000) loss: 2.302450
(Epoch 197 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 48301 / 49000) loss: 2.302417
(Iteration 48401 / 49000) loss: 2.302812
(Iteration 48501 / 49000) loss: 2.302750
(Epoch 198 / 200) train acc: 0.122000; val_acc: 0.087000
(Iteration 48601 / 49000) loss: 2.302578
(Iteration 48701 / 49000) loss: 2.302375
(Epoch 199 / 200) train acc: 0.118000; val_acc: 0.087000
(Iteration 48801 / 49000) loss: 2.302539
```

```
(Iteration 48901 / 49000) loss: 2.302547
(Epoch 200 / 200) train acc: 0.105000; val_acc: 0.087000
(Iteration 1 / 49000) loss: 519.478969
(Epoch 0 / 200) train acc: 0.086000; val_acc: 0.091000
(Iteration 101 / 49000) loss: 73.847950
(Iteration 201 / 49000) loss: 12.330019
(Epoch 1 / 200) train acc: 0.392000; val_acc: 0.378000
(Iteration 301 / 49000) loss: 3.752338
(Iteration 401 / 49000) loss: 2.518826
(Epoch 2 / 200) train acc: 0.178000; val_acc: 0.176000
(Iteration 501 / 49000) loss: 2.335601
(Iteration 601 / 49000) loss: 2.307717
(Iteration 701 / 49000) loss: 2.303284
(Epoch 3 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 801 / 49000) loss: 2.302488
(Iteration 901 / 49000) loss: 2.302733
(Epoch 4 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 1001 / 49000) loss: 2.303071
(Iteration 1101 / 49000) loss: 2.303169
(Iteration 1201 / 49000) loss: 2.302282
(Epoch 5 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 1301 / 49000) loss: 2.302866
(Iteration 1401 / 49000) loss: 2.301785
(Epoch 6 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 1501 / 49000) loss: 2.303654
(Iteration 1601 / 49000) loss: 2.302858
(Iteration 1701 / 49000) loss: 2.302241
(Epoch 7 / 200) train acc: 0.110000; val_acc: 0.079000
(Iteration 1801 / 49000) loss: 2.302598
(Iteration 1901 / 49000) loss: 2.302189
(Epoch 8 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 2001 / 49000) loss: 2.302133
(Iteration 2101 / 49000) loss: 2.303312
(Iteration 2201 / 49000) loss: 2.302017
(Epoch 9 / 200) train acc: 0.074000; val_acc: 0.079000
(Iteration 2301 / 49000) loss: 2.302530
(Iteration 2401 / 49000) loss: 2.300983
(Epoch 10 / 200) train acc: 0.119000; val_acc: 0.079000
(Iteration 2501 / 49000) loss: 2.302507
(Iteration 2601 / 49000) loss: 2.302435
(Epoch 11 / 200) train acc: 0.087000; val_acc: 0.079000
(Iteration 2701 / 49000) loss: 2.303572
(Iteration 2801 / 49000) loss: 2.302582
(Iteration 2901 / 49000) loss: 2.302261
(Epoch 12 / 200) train acc: 0.086000; val_acc: 0.087000
(Iteration 3001 / 49000) loss: 2.302401
(Iteration 3101 / 49000) loss: 2.302362
(Epoch 13 / 200) train acc: 0.096000; val_acc: 0.078000
```

```
(Iteration 3201 / 49000) loss: 2.302632
(Iteration 3301 / 49000) loss: 2.302713
(Iteration 3401 / 49000) loss: 2.302762
(Epoch 14 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 3501 / 49000) loss: 2.302477
(Iteration 3601 / 49000) loss: 2.302036
(Epoch 15 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 3701 / 49000) loss: 2.302435
(Iteration 3801 / 49000) loss: 2.302501
(Iteration 3901 / 49000) loss: 2.302926
(Epoch 16 / 200) train acc: 0.086000; val_acc: 0.087000
(Iteration 4001 / 49000) loss: 2.302389
(Iteration 4101 / 49000) loss: 2.303130
(Epoch 17 / 200) train acc: 0.111000; val_acc: 0.087000
(Iteration 4201 / 49000) loss: 2.302752
(Iteration 4301 / 49000) loss: 2.302793
(Iteration 4401 / 49000) loss: 2.301659
(Epoch 18 / 200) train acc: 0.087000; val_acc: 0.087000
(Iteration 4501 / 49000) loss: 2.303351
(Iteration 4601 / 49000) loss: 2.302597
(Epoch 19 / 200) train acc: 0.107000; val_acc: 0.087000
(Iteration 4701 / 49000) loss: 2.302715
(Iteration 4801 / 49000) loss: 2.302765
(Epoch 20 / 200) train acc: 0.111000; val_acc: 0.087000
(Iteration 4901 / 49000) loss: 2.302418
(Iteration 5001 / 49000) loss: 2.303204
(Iteration 5101 / 49000) loss: 2.302040
(Epoch 21 / 200) train acc: 0.086000; val_acc: 0.087000
(Iteration 5201 / 49000) loss: 2.302400
(Iteration 5301 / 49000) loss: 2.302369
(Epoch 22 / 200) train acc: 0.112000; val_acc: 0.079000
(Iteration 5401 / 49000) loss: 2.302822
(Iteration 5501 / 49000) loss: 2.302413
(Iteration 5601 / 49000) loss: 2.302593
(Epoch 23 / 200) train acc: 0.086000; val_acc: 0.079000
(Iteration 5701 / 49000) loss: 2.302626
(Iteration 5801 / 49000) loss: 2.302819
(Epoch 24 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 5901 / 49000) loss: 2.302336
(Iteration 6001 / 49000) loss: 2.302360
(Iteration 6101 / 49000) loss: 2.302565
(Epoch 25 / 200) train acc: 0.107000; val_acc: 0.105000
(Iteration 6201 / 49000) loss: 2.302364
(Iteration 6301 / 49000) loss: 2.302585
(Epoch 26 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 6401 / 49000) loss: 2.302320
(Iteration 6501 / 49000) loss: 2.302926
(Iteration 6601 / 49000) loss: 2.302642
```

```
(Epoch 27 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 6701 / 49000) loss: 2.302543
(Iteration 6801 / 49000) loss: 2.302291
(Epoch 28 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 6901 / 49000) loss: 2.302321
(Iteration 7001 / 49000) loss: 2.302387
(Iteration 7101 / 49000) loss: 2.302830
(Epoch 29 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 7201 / 49000) loss: 2.302810
(Iteration 7301 / 49000) loss: 2.301745
(Epoch 30 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 7401 / 49000) loss: 2.302822
(Iteration 7501 / 49000) loss: 2.302808
(Epoch 31 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 7601 / 49000) loss: 2.302584
(Iteration 7701 / 49000) loss: 2.302814
(Iteration 7801 / 49000) loss: 2.302393
(Epoch 32 / 200) train acc: 0.087000; val_acc: 0.078000
(Iteration 7901 / 49000) loss: 2.302704
(Iteration 8001 / 49000) loss: 2.302921
(Epoch 33 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 8101 / 49000) loss: 2.302119
(Iteration 8201 / 49000) loss: 2.302235
(Iteration 8301 / 49000) loss: 2.302102
(Epoch 34 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 8401 / 49000) loss: 2.302357
(Iteration 8501 / 49000) loss: 2.302597
(Epoch 35 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 8601 / 49000) loss: 2.303115
(Iteration 8701 / 49000) loss: 2.302950
(Iteration 8801 / 49000) loss: 2.302206
(Epoch 36 / 200) train acc: 0.092000; val_acc: 0.078000
(Iteration 8901 / 49000) loss: 2.302888
(Iteration 9001 / 49000) loss: 2.302365
(Epoch 37 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 9101 / 49000) loss: 2.302921
(Iteration 9201 / 49000) loss: 2.302882
(Iteration 9301 / 49000) loss: 2.303076
(Epoch 38 / 200) train acc: 0.117000; val_acc: 0.105000
(Iteration 9401 / 49000) loss: 2.302821
(Iteration 9501 / 49000) loss: 2.302692
(Epoch 39 / 200) train acc: 0.096000; val_acc: 0.102000
(Iteration 9601 / 49000) loss: 2.303049
(Iteration 9701 / 49000) loss: 2.302788
(Epoch 40 / 200) train acc: 0.105000; val_acc: 0.102000
(Iteration 9801 / 49000) loss: 2.302253
(Iteration 9901 / 49000) loss: 2.303036
(Iteration 10001 / 49000) loss: 2.302415
```

```
(Epoch 41 / 200) train acc: 0.090000; val_acc: 0.107000
(Iteration 10101 / 49000) loss: 2.301857
(Iteration 10201 / 49000) loss: 2.302899
(Epoch 42 / 200) train acc: 0.099000; val_acc: 0.105000
(Iteration 10301 / 49000) loss: 2.302547
(Iteration 10401 / 49000) loss: 2.302270
(Iteration 10501 / 49000) loss: 2.302233
(Epoch 43 / 200) train acc: 0.107000; val_acc: 0.105000
(Iteration 10601 / 49000) loss: 2.302507
(Iteration 10701 / 49000) loss: 2.302580
(Epoch 44 / 200) train acc: 0.104000; val_acc: 0.105000
(Iteration 10801 / 49000) loss: 2.302718
(Iteration 10901 / 49000) loss: 2.302518
(Iteration 11001 / 49000) loss: 2.302482
(Epoch 45 / 200) train acc: 0.091000; val_acc: 0.105000
(Iteration 11101 / 49000) loss: 2.302841
(Iteration 11201 / 49000) loss: 2.302325
(Epoch 46 / 200) train acc: 0.093000; val_acc: 0.105000
(Iteration 11301 / 49000) loss: 2.302399
(Iteration 11401 / 49000) loss: 2.302912
(Iteration 11501 / 49000) loss: 2.302191
(Epoch 47 / 200) train acc: 0.093000; val_acc: 0.105000
(Iteration 11601 / 49000) loss: 2.302165
(Iteration 11701 / 49000) loss: 2.302299
(Epoch 48 / 200) train acc: 0.098000; val_acc: 0.107000
(Iteration 11801 / 49000) loss: 2.302222
(Iteration 11901 / 49000) loss: 2.302957
(Iteration 12001 / 49000) loss: 2.302468
(Epoch 49 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 12101 / 49000) loss: 2.302435
(Iteration 12201 / 49000) loss: 2.302308
(Epoch 50 / 200) train acc: 0.099000; val_acc: 0.078000
(Iteration 12301 / 49000) loss: 2.302516
(Iteration 12401 / 49000) loss: 2.302887
(Epoch 51 / 200) train acc: 0.097000; val_acc: 0.105000
(Iteration 12501 / 49000) loss: 2.302338
(Iteration 12601 / 49000) loss: 2.302390
(Iteration 12701 / 49000) loss: 2.302848
(Epoch 52 / 200) train acc: 0.119000; val_acc: 0.078000
(Iteration 12801 / 49000) loss: 2.302210
(Iteration 12901 / 49000) loss: 2.302728
(Epoch 53 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 13001 / 49000) loss: 2.302614
(Iteration 13101 / 49000) loss: 2.302720
(Iteration 13201 / 49000) loss: 2.302458
(Epoch 54 / 200) train acc: 0.112000; val_acc: 0.078000
(Iteration 13301 / 49000) loss: 2.302547
(Iteration 13401 / 49000) loss: 2.302803
```

```
(Epoch 55 / 200) train acc: 0.092000; val_acc: 0.078000
(Iteration 13501 / 49000) loss: 2.302280
(Iteration 13601 / 49000) loss: 2.302401
(Iteration 13701 / 49000) loss: 2.302583
(Epoch 56 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 13801 / 49000) loss: 2.302713
(Iteration 13901 / 49000) loss: 2.303158
(Epoch 57 / 200) train acc: 0.107000; val_acc: 0.087000
(Iteration 14001 / 49000) loss: 2.302600
(Iteration 14101 / 49000) loss: 2.302143
(Iteration 14201 / 49000) loss: 2.302898
(Epoch 58 / 200) train acc: 0.096000; val_acc: 0.087000
(Iteration 14301 / 49000) loss: 2.302428
(Iteration 14401 / 49000) loss: 2.302632
(Epoch 59 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 14501 / 49000) loss: 2.302550
(Iteration 14601 / 49000) loss: 2.302871
(Epoch 60 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 14701 / 49000) loss: 2.302653
(Iteration 14801 / 49000) loss: 2.302194
(Iteration 14901 / 49000) loss: 2.301936
(Epoch 61 / 200) train acc: 0.093000; val_acc: 0.087000
(Iteration 15001 / 49000) loss: 2.302285
(Iteration 15101 / 49000) loss: 2.302124
(Epoch 62 / 200) train acc: 0.114000; val_acc: 0.087000
(Iteration 15201 / 49000) loss: 2.303200
(Iteration 15301 / 49000) loss: 2.302986
(Iteration 15401 / 49000) loss: 2.302996
(Epoch 63 / 200) train acc: 0.087000; val_acc: 0.087000
(Iteration 15501 / 49000) loss: 2.302455
(Iteration 15601 / 49000) loss: 2.302656
(Epoch 64 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 15701 / 49000) loss: 2.302542
(Iteration 15801 / 49000) loss: 2.302435
(Iteration 15901 / 49000) loss: 2.302145
(Epoch 65 / 200) train acc: 0.100000; val_acc: 0.078000
(Iteration 16001 / 49000) loss: 2.302578
(Iteration 16101 / 49000) loss: 2.302356
(Epoch 66 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 16201 / 49000) loss: 2.302922
(Iteration 16301 / 49000) loss: 2.302008
(Iteration 16401 / 49000) loss: 2.302304
(Epoch 67 / 200) train acc: 0.114000; val_acc: 0.079000
(Iteration 16501 / 49000) loss: 2.302219
(Iteration 16601 / 49000) loss: 2.302516
(Epoch 68 / 200) train acc: 0.093000; val_acc: 0.087000
(Iteration 16701 / 49000) loss: 2.302471
(Iteration 16801 / 49000) loss: 2.302767
```

```
(Iteration 16901 / 49000) loss: 2.302937
(Epoch 69 / 200) train acc: 0.107000; val_acc: 0.098000
(Iteration 17001 / 49000) loss: 2.302860
(Iteration 17101 / 49000) loss: 2.302435
(Epoch 70 / 200) train acc: 0.104000; val_acc: 0.098000
(Iteration 17201 / 49000) loss: 2.302813
(Iteration 17301 / 49000) loss: 2.302792
(Epoch 71 / 200) train acc: 0.087000; val_acc: 0.098000
(Iteration 17401 / 49000) loss: 2.302646
(Iteration 17501 / 49000) loss: 2.302681
(Iteration 17601 / 49000) loss: 2.302657
(Epoch 72 / 200) train acc: 0.090000; val_acc: 0.098000
(Iteration 17701 / 49000) loss: 2.302562
(Iteration 17801 / 49000) loss: 2.302753
(Epoch 73 / 200) train acc: 0.108000; val_acc: 0.098000
(Iteration 17901 / 49000) loss: 2.302743
(Iteration 18001 / 49000) loss: 2.302172
(Iteration 18101 / 49000) loss: 2.302593
(Epoch 74 / 200) train acc: 0.085000; val_acc: 0.098000
(Iteration 18201 / 49000) loss: 2.302673
(Iteration 18301 / 49000) loss: 2.302494
(Epoch 75 / 200) train acc: 0.112000; val_acc: 0.098000
(Iteration 18401 / 49000) loss: 2.302926
(Iteration 18501 / 49000) loss: 2.302597
(Iteration 18601 / 49000) loss: 2.302821
(Epoch 76 / 200) train acc: 0.096000; val_acc: 0.098000
(Iteration 18701 / 49000) loss: 2.302465
(Iteration 18801 / 49000) loss: 2.302667
(Epoch 77 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 18901 / 49000) loss: 2.302756
(Iteration 19001 / 49000) loss: 2.301784
(Iteration 19101 / 49000) loss: 2.302081
(Epoch 78 / 200) train acc: 0.099000; val_acc: 0.087000
(Iteration 19201 / 49000) loss: 2.302760
(Iteration 19301 / 49000) loss: 2.301920
(Epoch 79 / 200) train acc: 0.109000; val_acc: 0.087000
(Iteration 19401 / 49000) loss: 2.302350
(Iteration 19501 / 49000) loss: 2.303038
(Epoch 80 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 19601 / 49000) loss: 2.302896
(Iteration 19701 / 49000) loss: 2.302811
(Iteration 19801 / 49000) loss: 2.302994
(Epoch 81 / 200) train acc: 0.101000; val_acc: 0.098000
(Iteration 19901 / 49000) loss: 2.302857
(Iteration 20001 / 49000) loss: 2.302237
(Epoch 82 / 200) train acc: 0.106000; val_acc: 0.098000
(Iteration 20101 / 49000) loss: 2.302402
(Iteration 20201 / 49000) loss: 2.302955
```

```
(Iteration 20301 / 49000) loss: 2.302268
(Epoch 83 / 200) train acc: 0.098000; val_acc: 0.098000
(Iteration 20401 / 49000) loss: 2.302478
(Iteration 20501 / 49000) loss: 2.302653
(Epoch 84 / 200) train acc: 0.107000; val_acc: 0.098000
(Iteration 20601 / 49000) loss: 2.302501
(Iteration 20701 / 49000) loss: 2.302678
(Iteration 20801 / 49000) loss: 2.302856
(Epoch 85 / 200) train acc: 0.104000; val_acc: 0.098000
(Iteration 20901 / 49000) loss: 2.303088
(Iteration 21001 / 49000) loss: 2.302675
(Epoch 86 / 200) train acc: 0.102000; val_acc: 0.098000
(Iteration 21101 / 49000) loss: 2.303188
(Iteration 21201 / 49000) loss: 2.303114
(Iteration 21301 / 49000) loss: 2.302692
(Epoch 87 / 200) train acc: 0.108000; val_acc: 0.098000
(Iteration 21401 / 49000) loss: 2.302580
(Iteration 21501 / 49000) loss: 2.302242
(Epoch 88 / 200) train acc: 0.102000; val_acc: 0.098000
(Iteration 21601 / 49000) loss: 2.302671
(Iteration 21701 / 49000) loss: 2.302351
(Iteration 21801 / 49000) loss: 2.302850
(Epoch 89 / 200) train acc: 0.100000; val_acc: 0.098000
(Iteration 21901 / 49000) loss: 2.303208
(Iteration 22001 / 49000) loss: 2.302293
(Epoch 90 / 200) train acc: 0.077000; val_acc: 0.098000
(Iteration 22101 / 49000) loss: 2.302516
(Iteration 22201 / 49000) loss: 2.302484
(Epoch 91 / 200) train acc: 0.106000; val_acc: 0.098000
(Iteration 22301 / 49000) loss: 2.302669
(Iteration 22401 / 49000) loss: 2.302478
(Iteration 22501 / 49000) loss: 2.302278
(Epoch 92 / 200) train acc: 0.096000; val_acc: 0.098000
(Iteration 22601 / 49000) loss: 2.302563
(Iteration 22701 / 49000) loss: 2.302329
(Epoch 93 / 200) train acc: 0.101000; val_acc: 0.098000
(Iteration 22801 / 49000) loss: 2.302313
(Iteration 22901 / 49000) loss: 2.302883
(Iteration 23001 / 49000) loss: 2.302486
(Epoch 94 / 200) train acc: 0.091000; val_acc: 0.078000
(Iteration 23101 / 49000) loss: 2.302037
(Iteration 23201 / 49000) loss: 2.302224
(Epoch 95 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 23301 / 49000) loss: 2.302101
(Iteration 23401 / 49000) loss: 2.302624
(Iteration 23501 / 49000) loss: 2.302940
(Epoch 96 / 200) train acc: 0.077000; val_acc: 0.098000
(Iteration 23601 / 49000) loss: 2.302534
```

```
(Iteration 23701 / 49000) loss: 2.302820
(Epoch 97 / 200) train acc: 0.115000; val_acc: 0.098000
(Iteration 23801 / 49000) loss: 2.302857
(Iteration 23901 / 49000) loss: 2.302561
(Iteration 24001 / 49000) loss: 2.302604
(Epoch 98 / 200) train acc: 0.099000; val_acc: 0.098000
(Iteration 24101 / 49000) loss: 2.302774
(Iteration 24201 / 49000) loss: 2.302440
(Epoch 99 / 200) train acc: 0.107000; val_acc: 0.098000
(Iteration 24301 / 49000) loss: 2.302322
(Iteration 24401 / 49000) loss: 2.302611
(Epoch 100 / 200) train acc: 0.094000; val_acc: 0.098000
(Iteration 24501 / 49000) loss: 2.302557
(Iteration 24601 / 49000) loss: 2.302340
(Iteration 24701 / 49000) loss: 2.302091
(Epoch 101 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 24801 / 49000) loss: 2.302266
(Iteration 24901 / 49000) loss: 2.302430
(Epoch 102 / 200) train acc: 0.113000; val_acc: 0.098000
(Iteration 25001 / 49000) loss: 2.302608
(Iteration 25101 / 49000) loss: 2.302868
(Iteration 25201 / 49000) loss: 2.302150
(Epoch 103 / 200) train acc: 0.105000; val_acc: 0.098000
(Iteration 25301 / 49000) loss: 2.302314
(Iteration 25401 / 49000) loss: 2.302454
(Epoch 104 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 25501 / 49000) loss: 2.302571
(Iteration 25601 / 49000) loss: 2.302340
(Iteration 25701 / 49000) loss: 2.302395
(Epoch 105 / 200) train acc: 0.120000; val_acc: 0.079000
(Iteration 25801 / 49000) loss: 2.302465
(Iteration 25901 / 49000) loss: 2.302179
(Epoch 106 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 26001 / 49000) loss: 2.302416
(Iteration 26101 / 49000) loss: 2.302663
(Iteration 26201 / 49000) loss: 2.302687
(Epoch 107 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 26301 / 49000) loss: 2.302680
(Iteration 26401 / 49000) loss: 2.302937
(Epoch 108 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 26501 / 49000) loss: 2.302713
(Iteration 26601 / 49000) loss: 2.302641
(Iteration 26701 / 49000) loss: 2.302443
(Epoch 109 / 200) train acc: 0.110000; val_acc: 0.079000
(Iteration 26801 / 49000) loss: 2.302233
(Iteration 26901 / 49000) loss: 2.302299
(Epoch 110 / 200) train acc: 0.092000; val_acc: 0.079000
(Iteration 27001 / 49000) loss: 2.302712
```

```
(Iteration 27101 / 49000) loss: 2.302402
(Epoch 111 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 27201 / 49000) loss: 2.302637
(Iteration 27301 / 49000) loss: 2.302441
(Iteration 27401 / 49000) loss: 2.302406
(Epoch 112 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 27501 / 49000) loss: 2.302667
(Iteration 27601 / 49000) loss: 2.302772
(Epoch 113 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 27701 / 49000) loss: 2.302454
(Iteration 27801 / 49000) loss: 2.302828
(Iteration 27901 / 49000) loss: 2.302174
(Epoch 114 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 28001 / 49000) loss: 2.302624
(Iteration 28101 / 49000) loss: 2.302531
(Epoch 115 / 200) train acc: 0.119000; val_acc: 0.079000
(Iteration 28201 / 49000) loss: 2.302674
(Iteration 28301 / 49000) loss: 2.302828
(Iteration 28401 / 49000) loss: 2.302448
(Epoch 116 / 200) train acc: 0.088000; val_acc: 0.079000
(Iteration 28501 / 49000) loss: 2.303008
(Iteration 28601 / 49000) loss: 2.302426
(Epoch 117 / 200) train acc: 0.086000; val_acc: 0.079000
(Iteration 28701 / 49000) loss: 2.302933
(Iteration 28801 / 49000) loss: 2.302931
(Iteration 28901 / 49000) loss: 2.302248
(Epoch 118 / 200) train acc: 0.110000; val_acc: 0.079000
(Iteration 29001 / 49000) loss: 2.302906
(Iteration 29101 / 49000) loss: 2.302362
(Epoch 119 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 29201 / 49000) loss: 2.302532
(Iteration 29301 / 49000) loss: 2.302741
(Epoch 120 / 200) train acc: 0.083000; val_acc: 0.079000
(Iteration 29401 / 49000) loss: 2.302331
(Iteration 29501 / 49000) loss: 2.302330
(Iteration 29601 / 49000) loss: 2.302633
(Epoch 121 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 29701 / 49000) loss: 2.302458
(Iteration 29801 / 49000) loss: 2.302758
(Epoch 122 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 29901 / 49000) loss: 2.302400
(Iteration 30001 / 49000) loss: 2.302741
(Iteration 30101 / 49000) loss: 2.302491
(Epoch 123 / 200) train acc: 0.123000; val_acc: 0.079000
(Iteration 30201 / 49000) loss: 2.302315
(Iteration 30301 / 49000) loss: 2.302738
(Epoch 124 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 30401 / 49000) loss: 2.302861
```

```
(Iteration 30501 / 49000) loss: 2.302345
(Iteration 30601 / 49000) loss: 2.302572
(Epoch 125 / 200) train acc: 0.095000; val_acc: 0.078000
(Iteration 30701 / 49000) loss: 2.302776
(Iteration 30801 / 49000) loss: 2.302969
(Epoch 126 / 200) train acc: 0.114000; val_acc: 0.079000
(Iteration 30901 / 49000) loss: 2.302321
(Iteration 31001 / 49000) loss: 2.302639
(Iteration 31101 / 49000) loss: 2.302624
(Epoch 127 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 31201 / 49000) loss: 2.302571
(Iteration 31301 / 49000) loss: 2.302954
(Epoch 128 / 200) train acc: 0.092000; val_acc: 0.079000
(Iteration 31401 / 49000) loss: 2.302586
(Iteration 31501 / 49000) loss: 2.302675
(Iteration 31601 / 49000) loss: 2.302294
(Epoch 129 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 31701 / 49000) loss: 2.302623
(Iteration 31801 / 49000) loss: 2.303113
(Epoch 130 / 200) train acc: 0.085000; val_acc: 0.079000
(Iteration 31901 / 49000) loss: 2.302431
(Iteration 32001 / 49000) loss: 2.302443
(Epoch 131 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 32101 / 49000) loss: 2.302718
(Iteration 32201 / 49000) loss: 2.302543
(Iteration 32301 / 49000) loss: 2.302254
(Epoch 132 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 32401 / 49000) loss: 2.302449
(Iteration 32501 / 49000) loss: 2.302364
(Epoch 133 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 32601 / 49000) loss: 2.302226
(Iteration 32701 / 49000) loss: 2.302468
(Iteration 32801 / 49000) loss: 2.302489
(Epoch 134 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 32901 / 49000) loss: 2.302990
(Iteration 33001 / 49000) loss: 2.302347
(Epoch 135 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 33101 / 49000) loss: 2.302802
(Iteration 33201 / 49000) loss: 2.302390
(Iteration 33301 / 49000) loss: 2.302612
(Epoch 136 / 200) train acc: 0.118000; val_acc: 0.079000
(Iteration 33401 / 49000) loss: 2.302785
(Iteration 33501 / 49000) loss: 2.302495
(Epoch 137 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 33601 / 49000) loss: 2.302707
(Iteration 33701 / 49000) loss: 2.302430
(Iteration 33801 / 49000) loss: 2.302747
(Epoch 138 / 200) train acc: 0.091000; val_acc: 0.079000
```

```
(Iteration 33901 / 49000) loss: 2.302402
(Iteration 34001 / 49000) loss: 2.302678
(Epoch 139 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 34101 / 49000) loss: 2.302432
(Iteration 34201 / 49000) loss: 2.302579
(Epoch 140 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 34301 / 49000) loss: 2.302343
(Iteration 34401 / 49000) loss: 2.302545
(Iteration 34501 / 49000) loss: 2.302548
(Epoch 141 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 34601 / 49000) loss: 2.302432
(Iteration 34701 / 49000) loss: 2.302772
(Epoch 142 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 34801 / 49000) loss: 2.302425
(Iteration 34901 / 49000) loss: 2.302682
(Iteration 35001 / 49000) loss: 2.302432
(Epoch 143 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 35101 / 49000) loss: 2.302415
(Iteration 35201 / 49000) loss: 2.302291
(Epoch 144 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 35301 / 49000) loss: 2.302735
(Iteration 35401 / 49000) loss: 2.302470
(Iteration 35501 / 49000) loss: 2.302718
(Epoch 145 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 35601 / 49000) loss: 2.302681
(Iteration 35701 / 49000) loss: 2.302575
(Epoch 146 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 35801 / 49000) loss: 2.302716
(Iteration 35901 / 49000) loss: 2.303030
(Iteration 36001 / 49000) loss: 2.302567
(Epoch 147 / 200) train acc: 0.088000; val_acc: 0.079000
(Iteration 36101 / 49000) loss: 2.302534
(Iteration 36201 / 49000) loss: 2.302394
(Epoch 148 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 36301 / 49000) loss: 2.302390
(Iteration 36401 / 49000) loss: 2.302305
(Iteration 36501 / 49000) loss: 2.302796
(Epoch 149 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 36601 / 49000) loss: 2.302303
(Iteration 36701 / 49000) loss: 2.302485
(Epoch 150 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 36801 / 49000) loss: 2.302685
(Iteration 36901 / 49000) loss: 2.302889
(Epoch 151 / 200) train acc: 0.117000; val_acc: 0.079000
(Iteration 37001 / 49000) loss: 2.302714
(Iteration 37101 / 49000) loss: 2.302502
(Iteration 37201 / 49000) loss: 2.302489
(Epoch 152 / 200) train acc: 0.085000; val_acc: 0.079000
```

```
(Iteration 37301 / 49000) loss: 2.302817
(Iteration 37401 / 49000) loss: 2.302672
(Epoch 153 / 200) train acc: 0.113000; val_acc: 0.079000
(Iteration 37501 / 49000) loss: 2.302812
(Iteration 37601 / 49000) loss: 2.302660
(Iteration 37701 / 49000) loss: 2.302468
(Epoch 154 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 37801 / 49000) loss: 2.302684
(Iteration 37901 / 49000) loss: 2.302521
(Epoch 155 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 38001 / 49000) loss: 2.302489
(Iteration 38101 / 49000) loss: 2.302364
(Iteration 38201 / 49000) loss: 2.302335
(Epoch 156 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 38301 / 49000) loss: 2.302638
(Iteration 38401 / 49000) loss: 2.302623
(Epoch 157 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 38501 / 49000) loss: 2.302839
(Iteration 38601 / 49000) loss: 2.302571
(Iteration 38701 / 49000) loss: 2.302761
(Epoch 158 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 38801 / 49000) loss: 2.302832
(Iteration 38901 / 49000) loss: 2.302816
(Epoch 159 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 39001 / 49000) loss: 2.302842
(Iteration 39101 / 49000) loss: 2.302358
(Epoch 160 / 200) train acc: 0.088000; val_acc: 0.079000
(Iteration 39201 / 49000) loss: 2.302643
(Iteration 39301 / 49000) loss: 2.302656
(Iteration 39401 / 49000) loss: 2.302419
(Epoch 161 / 200) train acc: 0.084000; val_acc: 0.079000
(Iteration 39501 / 49000) loss: 2.302715
(Iteration 39601 / 49000) loss: 2.302871
(Epoch 162 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 39701 / 49000) loss: 2.302757
(Iteration 39801 / 49000) loss: 2.302559
(Iteration 39901 / 49000) loss: 2.302349
(Epoch 163 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 40001 / 49000) loss: 2.302629
(Iteration 40101 / 49000) loss: 2.302333
(Epoch 164 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 40201 / 49000) loss: 2.302834
(Iteration 40301 / 49000) loss: 2.302415
(Iteration 40401 / 49000) loss: 2.302707
(Epoch 165 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 40501 / 49000) loss: 2.303084
(Iteration 40601 / 49000) loss: 2.302842
(Epoch 166 / 200) train acc: 0.096000; val_acc: 0.079000
```

```
(Iteration 40701 / 49000) loss: 2.302413
(Iteration 40801 / 49000) loss: 2.302610
(Iteration 40901 / 49000) loss: 2.302575
(Epoch 167 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 41001 / 49000) loss: 2.302676
(Iteration 41101 / 49000) loss: 2.302766
(Epoch 168 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 41201 / 49000) loss: 2.302656
(Iteration 41301 / 49000) loss: 2.302568
(Iteration 41401 / 49000) loss: 2.302616
(Epoch 169 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 41501 / 49000) loss: 2.302591
(Iteration 41601 / 49000) loss: 2.302521
(Epoch 170 / 200) train acc: 0.086000; val_acc: 0.079000
(Iteration 41701 / 49000) loss: 2.302327
(Iteration 41801 / 49000) loss: 2.302674
(Epoch 171 / 200) train acc: 0.089000; val_acc: 0.079000
(Iteration 41901 / 49000) loss: 2.302719
(Iteration 42001 / 49000) loss: 2.302615
(Iteration 42101 / 49000) loss: 2.302739
(Epoch 172 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 42201 / 49000) loss: 2.302474
(Iteration 42301 / 49000) loss: 2.302956
(Epoch 173 / 200) train acc: 0.112000; val_acc: 0.079000
(Iteration 42401 / 49000) loss: 2.302718
(Iteration 42501 / 49000) loss: 2.302340
(Iteration 42601 / 49000) loss: 2.302598
(Epoch 174 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 42701 / 49000) loss: 2.302742
(Iteration 42801 / 49000) loss: 2.302584
(Epoch 175 / 200) train acc: 0.085000; val_acc: 0.079000
(Iteration 42901 / 49000) loss: 2.302711
(Iteration 43001 / 49000) loss: 2.302534
(Iteration 43101 / 49000) loss: 2.302193
(Epoch 176 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 43201 / 49000) loss: 2.302491
(Iteration 43301 / 49000) loss: 2.302440
(Epoch 177 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 43401 / 49000) loss: 2.302578
(Iteration 43501 / 49000) loss: 2.302414
(Iteration 43601 / 49000) loss: 2.302524
(Epoch 178 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 43701 / 49000) loss: 2.302638
(Iteration 43801 / 49000) loss: 2.302410
(Epoch 179 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 43901 / 49000) loss: 2.302445
(Iteration 44001 / 49000) loss: 2.302759
(Epoch 180 / 200) train acc: 0.120000; val_acc: 0.079000
```

```
(Iteration 44101 / 49000) loss: 2.302780
(Iteration 44201 / 49000) loss: 2.302406
(Iteration 44301 / 49000) loss: 2.302603
(Epoch 181 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 44401 / 49000) loss: 2.302358
(Iteration 44501 / 49000) loss: 2.302491
(Epoch 182 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 44601 / 49000) loss: 2.302326
(Iteration 44701 / 49000) loss: 2.302254
(Iteration 44801 / 49000) loss: 2.302389
(Epoch 183 / 200) train acc: 0.114000; val_acc: 0.079000
(Iteration 44901 / 49000) loss: 2.302774
(Iteration 45001 / 49000) loss: 2.302833
(Epoch 184 / 200) train acc: 0.117000; val_acc: 0.079000
(Iteration 45101 / 49000) loss: 2.302960
(Iteration 45201 / 49000) loss: 2.302403
(Iteration 45301 / 49000) loss: 2.301986
(Epoch 185 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 45401 / 49000) loss: 2.302480
(Iteration 45501 / 49000) loss: 2.302510
(Epoch 186 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 45601 / 49000) loss: 2.302717
(Iteration 45701 / 49000) loss: 2.302894
(Iteration 45801 / 49000) loss: 2.302514
(Epoch 187 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 45901 / 49000) loss: 2.302237
(Iteration 46001 / 49000) loss: 2.302462
(Epoch 188 / 200) train acc: 0.081000; val_acc: 0.079000
(Iteration 46101 / 49000) loss: 2.302373
(Iteration 46201 / 49000) loss: 2.302783
(Iteration 46301 / 49000) loss: 2.302428
(Epoch 189 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 46401 / 49000) loss: 2.302527
(Iteration 46501 / 49000) loss: 2.302975
(Epoch 190 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 46601 / 49000) loss: 2.302671
(Iteration 46701 / 49000) loss: 2.302746
(Epoch 191 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 46801 / 49000) loss: 2.302607
(Iteration 46901 / 49000) loss: 2.302544
(Iteration 47001 / 49000) loss: 2.302640
(Epoch 192 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 47101 / 49000) loss: 2.302672
(Iteration 47201 / 49000) loss: 2.302888
(Epoch 193 / 200) train acc: 0.075000; val_acc: 0.079000
(Iteration 47301 / 49000) loss: 2.302352
(Iteration 47401 / 49000) loss: 2.302686
(Iteration 47501 / 49000) loss: 2.302437
```

```
(Epoch 194 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 47601 / 49000) loss: 2.302630
(Iteration 47701 / 49000) loss: 2.302586
(Epoch 195 / 200) train acc: 0.120000; val_acc: 0.079000
(Iteration 47801 / 49000) loss: 2.302502
(Iteration 47901 / 49000) loss: 2.302620
(Iteration 48001 / 49000) loss: 2.302608
(Epoch 196 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 48101 / 49000) loss: 2.302732
(Iteration 48201 / 49000) loss: 2.302658
(Epoch 197 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 48301 / 49000) loss: 2.302773
(Iteration 48401 / 49000) loss: 2.302559
(Iteration 48501 / 49000) loss: 2.302596
(Epoch 198 / 200) train acc: 0.084000; val_acc: 0.079000
(Iteration 48601 / 49000) loss: 2.302614
(Iteration 48701 / 49000) loss: 2.302543
(Epoch 199 / 200) train acc: 0.082000; val_acc: 0.079000
(Iteration 48801 / 49000) loss: 2.302714
(Iteration 48901 / 49000) loss: 2.302770
(Epoch 200 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 1 / 49000) loss: 528.566156
(Epoch 0 / 200) train acc: 0.100000; val_acc: 0.114000
(Iteration 101 / 49000) loss: 3.586834
(Iteration 201 / 49000) loss: 2.305621
(Epoch 1 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 301 / 49000) loss: 2.303120
(Iteration 401 / 49000) loss: 2.302749
(Epoch 2 / 200) train acc: 0.098000; val_acc: 0.113000
(Iteration 501 / 49000) loss: 2.302604
(Iteration 601 / 49000) loss: 2.302846
(Iteration 701 / 49000) loss: 2.302827
(Epoch 3 / 200) train acc: 0.124000; val_acc: 0.105000
(Iteration 801 / 49000) loss: 2.302813
(Iteration 901 / 49000) loss: 2.302525
(Epoch 4 / 200) train acc: 0.083000; val_acc: 0.107000
(Iteration 1001 / 49000) loss: 2.303214
(Iteration 1101 / 49000) loss: 2.302298
(Iteration 1201 / 49000) loss: 2.302729
(Epoch 5 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 1301 / 49000) loss: 2.302301
(Iteration 1401 / 49000) loss: 2.303410
(Epoch 6 / 200) train acc: 0.096000; val_acc: 0.112000
(Iteration 1501 / 49000) loss: 2.302006
(Iteration 1601 / 49000) loss: 2.301283
(Iteration 1701 / 49000) loss: 2.302458
(Epoch 7 / 200) train acc: 0.097000; val_acc: 0.105000
(Iteration 1801 / 49000) loss: 2.302531
```

```
(Iteration 1901 / 49000) loss: 2.302375
(Epoch 8 / 200) train acc: 0.097000; val_acc: 0.105000
(Iteration 2001 / 49000) loss: 2.301643
(Iteration 2101 / 49000) loss: 2.304546
(Iteration 2201 / 49000) loss: 2.302279
(Epoch 9 / 200) train acc: 0.104000; val_acc: 0.113000
(Iteration 2301 / 49000) loss: 2.302510
(Iteration 2401 / 49000) loss: 2.303254
(Epoch 10 / 200) train acc: 0.111000; val_acc: 0.113000
(Iteration 2501 / 49000) loss: 2.302394
(Iteration 2601 / 49000) loss: 2.303011
(Epoch 11 / 200) train acc: 0.096000; val_acc: 0.112000
(Iteration 2701 / 49000) loss: 2.302137
(Iteration 2801 / 49000) loss: 2.302667
(Iteration 2901 / 49000) loss: 2.302052
(Epoch 12 / 200) train acc: 0.089000; val_acc: 0.079000
(Iteration 3001 / 49000) loss: 2.302169
(Iteration 3101 / 49000) loss: 2.302847
(Epoch 13 / 200) train acc: 0.082000; val_acc: 0.098000
(Iteration 3201 / 49000) loss: 2.302881
(Iteration 3301 / 49000) loss: 2.302414
(Iteration 3401 / 49000) loss: 2.302685
(Epoch 14 / 200) train acc: 0.099000; val_acc: 0.105000
(Iteration 3501 / 49000) loss: 2.303534
(Iteration 3601 / 49000) loss: 2.303530
(Epoch 15 / 200) train acc: 0.094000; val_acc: 0.098000
(Iteration 3701 / 49000) loss: 2.302058
(Iteration 3801 / 49000) loss: 2.301819
(Iteration 3901 / 49000) loss: 2.301854
(Epoch 16 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 4001 / 49000) loss: 2.302782
(Iteration 4101 / 49000) loss: 2.302651
(Epoch 17 / 200) train acc: 0.117000; val_acc: 0.079000
(Iteration 4201 / 49000) loss: 2.301582
(Iteration 4301 / 49000) loss: 2.302212
(Iteration 4401 / 49000) loss: 2.303224
(Epoch 18 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 4501 / 49000) loss: 2.304046
(Iteration 4601 / 49000) loss: 2.302910
(Epoch 19 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 4701 / 49000) loss: 2.302737
(Iteration 4801 / 49000) loss: 2.302530
(Epoch 20 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 4901 / 49000) loss: 2.303047
(Iteration 5001 / 49000) loss: 2.303006
(Iteration 5101 / 49000) loss: 2.302991
(Epoch 21 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 5201 / 49000) loss: 2.302712
```

```
(Iteration 5301 / 49000) loss: 2.302602
(Epoch 22 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 5401 / 49000) loss: 2.302783
(Iteration 5501 / 49000) loss: 2.303530
(Iteration 5601 / 49000) loss: 2.302631
(Epoch 23 / 200) train acc: 0.091000; val_acc: 0.098000
(Iteration 5701 / 49000) loss: 2.302416
(Iteration 5801 / 49000) loss: 2.301929
(Epoch 24 / 200) train acc: 0.091000; val_acc: 0.102000
(Iteration 5901 / 49000) loss: 2.302376
(Iteration 6001 / 49000) loss: 2.302132
(Iteration 6101 / 49000) loss: 2.303046
(Epoch 25 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 6201 / 49000) loss: 2.302584
(Iteration 6301 / 49000) loss: 2.302561
(Epoch 26 / 200) train acc: 0.087000; val_acc: 0.105000
(Iteration 6401 / 49000) loss: 2.302040
(Iteration 6501 / 49000) loss: 2.303203
(Iteration 6601 / 49000) loss: 2.302520
(Epoch 27 / 200) train acc: 0.100000; val_acc: 0.105000
(Iteration 6701 / 49000) loss: 2.302377
(Iteration 6801 / 49000) loss: 2.303134
(Epoch 28 / 200) train acc: 0.090000; val_acc: 0.098000
(Iteration 6901 / 49000) loss: 2.302265
(Iteration 7001 / 49000) loss: 2.302385
(Iteration 7101 / 49000) loss: 2.302245
(Epoch 29 / 200) train acc: 0.102000; val_acc: 0.105000
(Iteration 7201 / 49000) loss: 2.302395
(Iteration 7301 / 49000) loss: 2.302643
(Epoch 30 / 200) train acc: 0.095000; val_acc: 0.113000
(Iteration 7401 / 49000) loss: 2.302607
(Iteration 7501 / 49000) loss: 2.302567
(Epoch 31 / 200) train acc: 0.105000; val_acc: 0.113000
(Iteration 7601 / 49000) loss: 2.303069
(Iteration 7701 / 49000) loss: 2.302295
(Iteration 7801 / 49000) loss: 2.302228
(Epoch 32 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 7901 / 49000) loss: 2.302028
(Iteration 8001 / 49000) loss: 2.302529
(Epoch 33 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 8101 / 49000) loss: 2.302644
(Iteration 8201 / 49000) loss: 2.302197
(Iteration 8301 / 49000) loss: 2.303058
(Epoch 34 / 200) train acc: 0.105000; val_acc: 0.105000
(Iteration 8401 / 49000) loss: 2.302825
(Iteration 8501 / 49000) loss: 2.302805
(Epoch 35 / 200) train acc: 0.103000; val_acc: 0.105000
(Iteration 8601 / 49000) loss: 2.302200
```

```
(Iteration 8701 / 49000) loss: 2.302454
(Iteration 8801 / 49000) loss: 2.302836
(Epoch 36 / 200) train acc: 0.091000; val_acc: 0.105000
(Iteration 8901 / 49000) loss: 2.301987
(Iteration 9001 / 49000) loss: 2.302008
(Epoch 37 / 200) train acc: 0.097000; val_acc: 0.105000
(Iteration 9101 / 49000) loss: 2.301902
(Iteration 9201 / 49000) loss: 2.302583
(Iteration 9301 / 49000) loss: 2.302361
(Epoch 38 / 200) train acc: 0.108000; val_acc: 0.105000
(Iteration 9401 / 49000) loss: 2.302754
(Iteration 9501 / 49000) loss: 2.302375
(Epoch 39 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 9601 / 49000) loss: 2.302116
(Iteration 9701 / 49000) loss: 2.302910
(Epoch 40 / 200) train acc: 0.092000; val_acc: 0.105000
(Iteration 9801 / 49000) loss: 2.302096
(Iteration 9901 / 49000) loss: 2.301917
(Iteration 10001 / 49000) loss: 2.302612
(Epoch 41 / 200) train acc: 0.089000; val_acc: 0.105000
(Iteration 10101 / 49000) loss: 2.302619
(Iteration 10201 / 49000) loss: 2.302465
(Epoch 42 / 200) train acc: 0.096000; val_acc: 0.119000
(Iteration 10301 / 49000) loss: 2.301880
(Iteration 10401 / 49000) loss: 2.302609
(Iteration 10501 / 49000) loss: 2.302551
(Epoch 43 / 200) train acc: 0.117000; val_acc: 0.119000
(Iteration 10601 / 49000) loss: 2.302933
(Iteration 10701 / 49000) loss: 2.302571
(Epoch 44 / 200) train acc: 0.103000; val_acc: 0.119000
(Iteration 10801 / 49000) loss: 2.302916
(Iteration 10901 / 49000) loss: 2.302356
(Iteration 11001 / 49000) loss: 2.302737
(Epoch 45 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 11101 / 49000) loss: 2.302475
(Iteration 11201 / 49000) loss: 2.302239
(Epoch 46 / 200) train acc: 0.107000; val_acc: 0.102000
(Iteration 11301 / 49000) loss: 2.302577
(Iteration 11401 / 49000) loss: 2.302240
(Iteration 11501 / 49000) loss: 2.302843
(Epoch 47 / 200) train acc: 0.108000; val_acc: 0.102000
(Iteration 11601 / 49000) loss: 2.303555
(Iteration 11701 / 49000) loss: 2.302637
(Epoch 48 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 11801 / 49000) loss: 2.302034
(Iteration 11901 / 49000) loss: 2.302715
(Iteration 12001 / 49000) loss: 2.302414
(Epoch 49 / 200) train acc: 0.106000; val_acc: 0.087000
```

```
(Iteration 12101 / 49000) loss: 2.302935
(Iteration 12201 / 49000) loss: 2.302479
(Epoch 50 / 200) train acc: 0.111000; val_acc: 0.087000
(Iteration 12301 / 49000) loss: 2.302986
(Iteration 12401 / 49000) loss: 2.303198
(Epoch 51 / 200) train acc: 0.086000; val_acc: 0.078000
(Iteration 12501 / 49000) loss: 2.302877
(Iteration 12601 / 49000) loss: 2.302347
(Iteration 12701 / 49000) loss: 2.302579
(Epoch 52 / 200) train acc: 0.087000; val_acc: 0.102000
(Iteration 12801 / 49000) loss: 2.302168
(Iteration 12901 / 49000) loss: 2.302719
(Epoch 53 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 13001 / 49000) loss: 2.302497
(Iteration 13101 / 49000) loss: 2.302633
(Iteration 13201 / 49000) loss: 2.302692
(Epoch 54 / 200) train acc: 0.089000; val_acc: 0.102000
(Iteration 13301 / 49000) loss: 2.302016
(Iteration 13401 / 49000) loss: 2.303331
(Epoch 55 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 13501 / 49000) loss: 2.303038
(Iteration 13601 / 49000) loss: 2.301916
(Iteration 13701 / 49000) loss: 2.302409
(Epoch 56 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 13801 / 49000) loss: 2.301762
(Iteration 13901 / 49000) loss: 2.302036
(Epoch 57 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 14001 / 49000) loss: 2.302025
(Iteration 14101 / 49000) loss: 2.302338
(Iteration 14201 / 49000) loss: 2.302910
(Epoch 58 / 200) train acc: 0.096000; val_acc: 0.098000
(Iteration 14301 / 49000) loss: 2.302746
(Iteration 14401 / 49000) loss: 2.302622
(Epoch 59 / 200) train acc: 0.100000; val_acc: 0.098000
(Iteration 14501 / 49000) loss: 2.302154
(Iteration 14601 / 49000) loss: 2.302523
(Epoch 60 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 14701 / 49000) loss: 2.302242
(Iteration 14801 / 49000) loss: 2.302143
(Iteration 14901 / 49000) loss: 2.302978
(Epoch 61 / 200) train acc: 0.103000; val_acc: 0.102000
(Iteration 15001 / 49000) loss: 2.302734
(Iteration 15101 / 49000) loss: 2.302715
(Epoch 62 / 200) train acc: 0.111000; val_acc: 0.079000
(Iteration 15201 / 49000) loss: 2.302896
(Iteration 15301 / 49000) loss: 2.302139
(Iteration 15401 / 49000) loss: 2.302396
(Epoch 63 / 200) train acc: 0.108000; val_acc: 0.102000
```

```
(Iteration 15501 / 49000) loss: 2.302632
(Iteration 15601 / 49000) loss: 2.302628
(Epoch 64 / 200) train acc: 0.098000; val_acc: 0.102000
(Iteration 15701 / 49000) loss: 2.302928
(Iteration 15801 / 49000) loss: 2.302470
(Iteration 15901 / 49000) loss: 2.302711
(Epoch 65 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 16001 / 49000) loss: 2.302776
(Iteration 16101 / 49000) loss: 2.302394
(Epoch 66 / 200) train acc: 0.093000; val_acc: 0.102000
(Iteration 16201 / 49000) loss: 2.302344
(Iteration 16301 / 49000) loss: 2.302421
(Iteration 16401 / 49000) loss: 2.302602
(Epoch 67 / 200) train acc: 0.114000; val_acc: 0.087000
(Iteration 16501 / 49000) loss: 2.302711
(Iteration 16601 / 49000) loss: 2.302319
(Epoch 68 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 16701 / 49000) loss: 2.302478
(Iteration 16801 / 49000) loss: 2.302372
(Iteration 16901 / 49000) loss: 2.302571
(Epoch 69 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 17001 / 49000) loss: 2.302870
(Iteration 17101 / 49000) loss: 2.302481
(Epoch 70 / 200) train acc: 0.118000; val_acc: 0.087000
(Iteration 17201 / 49000) loss: 2.301798
(Iteration 17301 / 49000) loss: 2.302032
(Epoch 71 / 200) train acc: 0.099000; val_acc: 0.087000
(Iteration 17401 / 49000) loss: 2.302947
(Iteration 17501 / 49000) loss: 2.301987
(Iteration 17601 / 49000) loss: 2.302588
(Epoch 72 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 17701 / 49000) loss: 2.302948
(Iteration 17801 / 49000) loss: 2.302706
(Epoch 73 / 200) train acc: 0.096000; val_acc: 0.087000
(Iteration 17901 / 49000) loss: 2.302459
(Iteration 18001 / 49000) loss: 2.302764
(Iteration 18101 / 49000) loss: 2.302656
(Epoch 74 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 18201 / 49000) loss: 2.303210
(Iteration 18301 / 49000) loss: 2.302353
(Epoch 75 / 200) train acc: 0.084000; val_acc: 0.087000
(Iteration 18401 / 49000) loss: 2.302588
(Iteration 18501 / 49000) loss: 2.302885
(Iteration 18601 / 49000) loss: 2.302644
(Epoch 76 / 200) train acc: 0.096000; val_acc: 0.087000
(Iteration 18701 / 49000) loss: 2.302594
(Iteration 18801 / 49000) loss: 2.302591
(Epoch 77 / 200) train acc: 0.084000; val_acc: 0.087000
```

```
(Iteration 18901 / 49000) loss: 2.303054
(Iteration 19001 / 49000) loss: 2.302614
(Iteration 19101 / 49000) loss: 2.302927
(Epoch 78 / 200) train acc: 0.119000; val_acc: 0.078000
(Iteration 19201 / 49000) loss: 2.302775
(Iteration 19301 / 49000) loss: 2.302361
(Epoch 79 / 200) train acc: 0.111000; val_acc: 0.078000
(Iteration 19401 / 49000) loss: 2.302599
(Iteration 19501 / 49000) loss: 2.302799
(Epoch 80 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 19601 / 49000) loss: 2.303190
(Iteration 19701 / 49000) loss: 2.302368
(Iteration 19801 / 49000) loss: 2.302160
(Epoch 81 / 200) train acc: 0.087000; val_acc: 0.078000
(Iteration 19901 / 49000) loss: 2.302487
(Iteration 20001 / 49000) loss: 2.302579
(Epoch 82 / 200) train acc: 0.111000; val_acc: 0.078000
(Iteration 20101 / 49000) loss: 2.302734
(Iteration 20201 / 49000) loss: 2.303030
(Iteration 20301 / 49000) loss: 2.302946
(Epoch 83 / 200) train acc: 0.116000; val_acc: 0.078000
(Iteration 20401 / 49000) loss: 2.302569
(Iteration 20501 / 49000) loss: 2.302999
(Epoch 84 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 20601 / 49000) loss: 2.302864
(Iteration 20701 / 49000) loss: 2.301962
(Iteration 20801 / 49000) loss: 2.302604
(Epoch 85 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 20901 / 49000) loss: 2.302394
(Iteration 21001 / 49000) loss: 2.302693
(Epoch 86 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 21101 / 49000) loss: 2.302245
(Iteration 21201 / 49000) loss: 2.302411
(Iteration 21301 / 49000) loss: 2.302979
(Epoch 87 / 200) train acc: 0.099000; val_acc: 0.078000
(Iteration 21401 / 49000) loss: 2.302667
(Iteration 21501 / 49000) loss: 2.302767
(Epoch 88 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 21601 / 49000) loss: 2.302254
(Iteration 21701 / 49000) loss: 2.302182
(Iteration 21801 / 49000) loss: 2.302707
(Epoch 89 / 200) train acc: 0.115000; val_acc: 0.078000
(Iteration 21901 / 49000) loss: 2.302403
(Iteration 22001 / 49000) loss: 2.302591
(Epoch 90 / 200) train acc: 0.121000; val_acc: 0.078000
(Iteration 22101 / 49000) loss: 2.302386
(Iteration 22201 / 49000) loss: 2.302643
(Epoch 91 / 200) train acc: 0.108000; val_acc: 0.078000
```

```
(Iteration 22301 / 49000) loss: 2.302648
(Iteration 22401 / 49000) loss: 2.302472
(Iteration 22501 / 49000) loss: 2.302106
(Epoch 92 / 200) train acc: 0.083000; val_acc: 0.078000
(Iteration 22601 / 49000) loss: 2.302417
(Iteration 22701 / 49000) loss: 2.302773
(Epoch 93 / 200) train acc: 0.092000; val_acc: 0.078000
(Iteration 22801 / 49000) loss: 2.302544
(Iteration 22901 / 49000) loss: 2.302256
(Iteration 23001 / 49000) loss: 2.302584
(Epoch 94 / 200) train acc: 0.088000; val_acc: 0.079000
(Iteration 23101 / 49000) loss: 2.302540
(Iteration 23201 / 49000) loss: 2.302316
(Epoch 95 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 23301 / 49000) loss: 2.302360
(Iteration 23401 / 49000) loss: 2.303418
(Iteration 23501 / 49000) loss: 2.302836
(Epoch 96 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 23601 / 49000) loss: 2.302701
(Iteration 23701 / 49000) loss: 2.302770
(Epoch 97 / 200) train acc: 0.112000; val_acc: 0.079000
(Iteration 23801 / 49000) loss: 2.302790
(Iteration 23901 / 49000) loss: 2.302983
(Iteration 24001 / 49000) loss: 2.302712
(Epoch 98 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 24101 / 49000) loss: 2.302251
(Iteration 24201 / 49000) loss: 2.302535
(Epoch 99 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 24301 / 49000) loss: 2.302486
(Iteration 24401 / 49000) loss: 2.302343
(Epoch 100 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 24501 / 49000) loss: 2.302338
(Iteration 24601 / 49000) loss: 2.302572
(Iteration 24701 / 49000) loss: 2.302451
(Epoch 101 / 200) train acc: 0.089000; val_acc: 0.079000
(Iteration 24801 / 49000) loss: 2.302755
(Iteration 24901 / 49000) loss: 2.302301
(Epoch 102 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 25001 / 49000) loss: 2.302851
(Iteration 25101 / 49000) loss: 2.302800
(Iteration 25201 / 49000) loss: 2.302344
(Epoch 103 / 200) train acc: 0.085000; val_acc: 0.079000
(Iteration 25301 / 49000) loss: 2.302611
(Iteration 25401 / 49000) loss: 2.302723
(Epoch 104 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 25501 / 49000) loss: 2.302549
(Iteration 25601 / 49000) loss: 2.302646
(Iteration 25701 / 49000) loss: 2.302717
```

```
(Epoch 105 / 200) train acc: 0.086000; val_acc: 0.079000
(Iteration 25801 / 49000) loss: 2.302341
(Iteration 25901 / 49000) loss: 2.302173
(Epoch 106 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 26001 / 49000) loss: 2.302750
(Iteration 26101 / 49000) loss: 2.302659
(Iteration 26201 / 49000) loss: 2.302742
(Epoch 107 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 26301 / 49000) loss: 2.302498
(Iteration 26401 / 49000) loss: 2.302402
(Epoch 108 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 26501 / 49000) loss: 2.302233
(Iteration 26601 / 49000) loss: 2.302474
(Iteration 26701 / 49000) loss: 2.302719
(Epoch 109 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 26801 / 49000) loss: 2.302807
(Iteration 26901 / 49000) loss: 2.302434
(Epoch 110 / 200) train acc: 0.082000; val_acc: 0.079000
(Iteration 27001 / 49000) loss: 2.302166
(Iteration 27101 / 49000) loss: 2.302661
(Epoch 111 / 200) train acc: 0.092000; val_acc: 0.079000
(Iteration 27201 / 49000) loss: 2.302415
(Iteration 27301 / 49000) loss: 2.302436
(Iteration 27401 / 49000) loss: 2.301986
(Epoch 112 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 27501 / 49000) loss: 2.302460
(Iteration 27601 / 49000) loss: 2.302838
(Epoch 113 / 200) train acc: 0.083000; val_acc: 0.079000
(Iteration 27701 / 49000) loss: 2.303241
(Iteration 27801 / 49000) loss: 2.302293
(Iteration 27901 / 49000) loss: 2.302427
(Epoch 114 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 28001 / 49000) loss: 2.302455
(Iteration 28101 / 49000) loss: 2.302693
(Epoch 115 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 28201 / 49000) loss: 2.302855
(Iteration 28301 / 49000) loss: 2.302398
(Iteration 28401 / 49000) loss: 2.302859
(Epoch 116 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 28501 / 49000) loss: 2.302556
(Iteration 28601 / 49000) loss: 2.302991
(Epoch 117 / 200) train acc: 0.085000; val_acc: 0.079000
(Iteration 28701 / 49000) loss: 2.302321
(Iteration 28801 / 49000) loss: 2.302584
(Iteration 28901 / 49000) loss: 2.302841
(Epoch 118 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 29001 / 49000) loss: 2.302124
(Iteration 29101 / 49000) loss: 2.302561
```

```
(Epoch 119 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 29201 / 49000) loss: 2.302994
(Iteration 29301 / 49000) loss: 2.302386
(Epoch 120 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 29401 / 49000) loss: 2.302628
(Iteration 29501 / 49000) loss: 2.303146
(Iteration 29601 / 49000) loss: 2.301816
(Epoch 121 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 29701 / 49000) loss: 2.302993
(Iteration 29801 / 49000) loss: 2.302189
(Epoch 122 / 200) train acc: 0.110000; val_acc: 0.079000
(Iteration 29901 / 49000) loss: 2.301860
(Iteration 30001 / 49000) loss: 2.302998
(Iteration 30101 / 49000) loss: 2.303156
(Epoch 123 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 30201 / 49000) loss: 2.302336
(Iteration 30301 / 49000) loss: 2.302441
(Epoch 124 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 30401 / 49000) loss: 2.302799
(Iteration 30501 / 49000) loss: 2.302962
(Iteration 30601 / 49000) loss: 2.302969
(Epoch 125 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 30701 / 49000) loss: 2.302808
(Iteration 30801 / 49000) loss: 2.302756
(Epoch 126 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 30901 / 49000) loss: 2.302772
(Iteration 31001 / 49000) loss: 2.302043
(Iteration 31101 / 49000) loss: 2.302306
(Epoch 127 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 31201 / 49000) loss: 2.302051
(Iteration 31301 / 49000) loss: 2.302588
(Epoch 128 / 200) train acc: 0.111000; val_acc: 0.079000
(Iteration 31401 / 49000) loss: 2.302462
(Iteration 31501 / 49000) loss: 2.302638
(Iteration 31601 / 49000) loss: 2.302828
(Epoch 129 / 200) train acc: 0.089000; val_acc: 0.079000
(Iteration 31701 / 49000) loss: 2.302850
(Iteration 31801 / 49000) loss: 2.302263
(Epoch 130 / 200) train acc: 0.083000; val_acc: 0.079000
(Iteration 31901 / 49000) loss: 2.302214
(Iteration 32001 / 49000) loss: 2.302683
(Epoch 131 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 32101 / 49000) loss: 2.302953
(Iteration 32201 / 49000) loss: 2.302558
(Iteration 32301 / 49000) loss: 2.302606
(Epoch 132 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 32401 / 49000) loss: 2.302851
(Iteration 32501 / 49000) loss: 2.302196
```

```
(Epoch 133 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 32601 / 49000) loss: 2.302553
(Iteration 32701 / 49000) loss: 2.302481
(Iteration 32801 / 49000) loss: 2.302274
(Epoch 134 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 32901 / 49000) loss: 2.302632
(Iteration 33001 / 49000) loss: 2.302766
(Epoch 135 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 33101 / 49000) loss: 2.302799
(Iteration 33201 / 49000) loss: 2.302465
(Iteration 33301 / 49000) loss: 2.302509
(Epoch 136 / 200) train acc: 0.089000; val_acc: 0.079000
(Iteration 33401 / 49000) loss: 2.302879
(Iteration 33501 / 49000) loss: 2.302117
(Epoch 137 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 33601 / 49000) loss: 2.302249
(Iteration 33701 / 49000) loss: 2.302559
(Iteration 33801 / 49000) loss: 2.303103
(Epoch 138 / 200) train acc: 0.089000; val_acc: 0.079000
(Iteration 33901 / 49000) loss: 2.302073
(Iteration 34001 / 49000) loss: 2.302217
(Epoch 139 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 34101 / 49000) loss: 2.302537
(Iteration 34201 / 49000) loss: 2.302827
(Epoch 140 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 34301 / 49000) loss: 2.303003
(Iteration 34401 / 49000) loss: 2.302222
(Iteration 34501 / 49000) loss: 2.303107
(Epoch 141 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 34601 / 49000) loss: 2.302946
(Iteration 34701 / 49000) loss: 2.301915
(Epoch 142 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 34801 / 49000) loss: 2.302495
(Iteration 34901 / 49000) loss: 2.302304
(Iteration 35001 / 49000) loss: 2.302188
(Epoch 143 / 200) train acc: 0.118000; val_acc: 0.079000
(Iteration 35101 / 49000) loss: 2.303286
(Iteration 35201 / 49000) loss: 2.302639
(Epoch 144 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 35301 / 49000) loss: 2.302303
(Iteration 35401 / 49000) loss: 2.302546
(Iteration 35501 / 49000) loss: 2.302970
(Epoch 145 / 200) train acc: 0.083000; val_acc: 0.079000
(Iteration 35601 / 49000) loss: 2.302265
(Iteration 35701 / 49000) loss: 2.302439
(Epoch 146 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 35801 / 49000) loss: 2.302923
(Iteration 35901 / 49000) loss: 2.302514
```

```
(Iteration 36001 / 49000) loss: 2.302299
(Epoch 147 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 36101 / 49000) loss: 2.302468
(Iteration 36201 / 49000) loss: 2.302409
(Epoch 148 / 200) train acc: 0.116000; val_acc: 0.079000
(Iteration 36301 / 49000) loss: 2.302286
(Iteration 36401 / 49000) loss: 2.302457
(Iteration 36501 / 49000) loss: 2.302666
(Epoch 149 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 36601 / 49000) loss: 2.302525
(Iteration 36701 / 49000) loss: 2.302686
(Epoch 150 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 36801 / 49000) loss: 2.302540
(Iteration 36901 / 49000) loss: 2.303051
(Epoch 151 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 37001 / 49000) loss: 2.302302
(Iteration 37101 / 49000) loss: 2.302423
(Iteration 37201 / 49000) loss: 2.302806
(Epoch 152 / 200) train acc: 0.087000; val_acc: 0.079000
(Iteration 37301 / 49000) loss: 2.302491
(Iteration 37401 / 49000) loss: 2.302613
(Epoch 153 / 200) train acc: 0.112000; val_acc: 0.079000
(Iteration 37501 / 49000) loss: 2.302903
(Iteration 37601 / 49000) loss: 2.303040
(Iteration 37701 / 49000) loss: 2.302546
(Epoch 154 / 200) train acc: 0.129000; val_acc: 0.079000
(Iteration 37801 / 49000) loss: 2.302842
(Iteration 37901 / 49000) loss: 2.302321
(Epoch 155 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 38001 / 49000) loss: 2.302805
(Iteration 38101 / 49000) loss: 2.302688
(Iteration 38201 / 49000) loss: 2.302976
(Epoch 156 / 200) train acc: 0.083000; val_acc: 0.079000
(Iteration 38301 / 49000) loss: 2.302451
(Iteration 38401 / 49000) loss: 2.302761
(Epoch 157 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 38501 / 49000) loss: 2.302579
(Iteration 38601 / 49000) loss: 2.302486
(Iteration 38701 / 49000) loss: 2.302154
(Epoch 158 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 38801 / 49000) loss: 2.302234
(Iteration 38901 / 49000) loss: 2.302586
(Epoch 159 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 39001 / 49000) loss: 2.302571
(Iteration 39101 / 49000) loss: 2.302746
(Epoch 160 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 39201 / 49000) loss: 2.302722
(Iteration 39301 / 49000) loss: 2.302681
```

```
(Iteration 39401 / 49000) loss: 2.302347
(Epoch 161 / 200) train acc: 0.109000; val_acc: 0.078000
(Iteration 39501 / 49000) loss: 2.302665
(Iteration 39601 / 49000) loss: 2.302476
(Epoch 162 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 39701 / 49000) loss: 2.302332
(Iteration 39801 / 49000) loss: 2.302666
(Iteration 39901 / 49000) loss: 2.302483
(Epoch 163 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 40001 / 49000) loss: 2.302444
(Iteration 40101 / 49000) loss: 2.302713
(Epoch 164 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 40201 / 49000) loss: 2.302650
(Iteration 40301 / 49000) loss: 2.302599
(Iteration 40401 / 49000) loss: 2.302500
(Epoch 165 / 200) train acc: 0.100000; val_acc: 0.078000
(Iteration 40501 / 49000) loss: 2.302166
(Iteration 40601 / 49000) loss: 2.302695
(Epoch 166 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 40701 / 49000) loss: 2.302407
(Iteration 40801 / 49000) loss: 2.302781
(Iteration 40901 / 49000) loss: 2.302531
(Epoch 167 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 41001 / 49000) loss: 2.302454
(Iteration 41101 / 49000) loss: 2.302245
(Epoch 168 / 200) train acc: 0.089000; val_acc: 0.078000
(Iteration 41201 / 49000) loss: 2.302458
(Iteration 41301 / 49000) loss: 2.302731
(Iteration 41401 / 49000) loss: 2.302537
(Epoch 169 / 200) train acc: 0.109000; val_acc: 0.078000
(Iteration 41501 / 49000) loss: 2.302511
(Iteration 41601 / 49000) loss: 2.302686
(Epoch 170 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 41701 / 49000) loss: 2.302606
(Iteration 41801 / 49000) loss: 2.302401
(Epoch 171 / 200) train acc: 0.113000; val_acc: 0.078000
(Iteration 41901 / 49000) loss: 2.302250
(Iteration 42001 / 49000) loss: 2.302526
(Iteration 42101 / 49000) loss: 2.302229
(Epoch 172 / 200) train acc: 0.081000; val_acc: 0.078000
(Iteration 42201 / 49000) loss: 2.302903
(Iteration 42301 / 49000) loss: 2.302608
(Epoch 173 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 42401 / 49000) loss: 2.302897
(Iteration 42501 / 49000) loss: 2.302721
(Iteration 42601 / 49000) loss: 2.302690
(Epoch 174 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 42701 / 49000) loss: 2.302898
```

```
(Iteration 42801 / 49000) loss: 2.302869
(Epoch 175 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 42901 / 49000) loss: 2.302506
(Iteration 43001 / 49000) loss: 2.302423
(Iteration 43101 / 49000) loss: 2.302634
(Epoch 176 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 43201 / 49000) loss: 2.302387
(Iteration 43301 / 49000) loss: 2.302574
(Epoch 177 / 200) train acc: 0.123000; val_acc: 0.079000
(Iteration 43401 / 49000) loss: 2.302298
(Iteration 43501 / 49000) loss: 2.302897
(Iteration 43601 / 49000) loss: 2.302553
(Epoch 178 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 43701 / 49000) loss: 2.302824
(Iteration 43801 / 49000) loss: 2.302428
(Epoch 179 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 43901 / 49000) loss: 2.302332
(Iteration 44001 / 49000) loss: 2.302573
(Epoch 180 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 44101 / 49000) loss: 2.302726
(Iteration 44201 / 49000) loss: 2.302355
(Iteration 44301 / 49000) loss: 2.302707
(Epoch 181 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 44401 / 49000) loss: 2.302705
(Iteration 44501 / 49000) loss: 2.302774
(Epoch 182 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 44601 / 49000) loss: 2.302503
(Iteration 44701 / 49000) loss: 2.302544
(Iteration 44801 / 49000) loss: 2.302745
(Epoch 183 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 44901 / 49000) loss: 2.302801
(Iteration 45001 / 49000) loss: 2.302671
(Epoch 184 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 45101 / 49000) loss: 2.302627
(Iteration 45201 / 49000) loss: 2.302664
(Iteration 45301 / 49000) loss: 2.302622
(Epoch 185 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 45401 / 49000) loss: 2.302595
(Iteration 45501 / 49000) loss: 2.302622
(Epoch 186 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 45601 / 49000) loss: 2.302329
(Iteration 45701 / 49000) loss: 2.302375
(Iteration 45801 / 49000) loss: 2.302474
(Epoch 187 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 45901 / 49000) loss: 2.302770
(Iteration 46001 / 49000) loss: 2.302468
(Epoch 188 / 200) train acc: 0.087000; val_acc: 0.079000
(Iteration 46101 / 49000) loss: 2.302663
```

```
(Iteration 46201 / 49000) loss: 2.302547
(Iteration 46301 / 49000) loss: 2.302650
(Epoch 189 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 46401 / 49000) loss: 2.302431
(Iteration 46501 / 49000) loss: 2.302601
(Epoch 190 / 200) train acc: 0.109000; val_acc: 0.078000
(Iteration 46601 / 49000) loss: 2.302153
(Iteration 46701 / 49000) loss: 2.302596
(Epoch 191 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 46801 / 49000) loss: 2.302551
(Iteration 46901 / 49000) loss: 2.302493
(Iteration 47001 / 49000) loss: 2.302475
(Epoch 192 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 47101 / 49000) loss: 2.302698
(Iteration 47201 / 49000) loss: 2.302432
(Epoch 193 / 200) train acc: 0.111000; val_acc: 0.078000
(Iteration 47301 / 49000) loss: 2.302630
(Iteration 47401 / 49000) loss: 2.302613
(Iteration 47501 / 49000) loss: 2.302336
(Epoch 194 / 200) train acc: 0.089000; val_acc: 0.078000
(Iteration 47601 / 49000) loss: 2.302629
(Iteration 47701 / 49000) loss: 2.302302
(Epoch 195 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 47801 / 49000) loss: 2.302783
(Iteration 47901 / 49000) loss: 2.302561
(Iteration 48001 / 49000) loss: 2.302613
(Epoch 196 / 200) train acc: 0.088000; val_acc: 0.078000
(Iteration 48101 / 49000) loss: 2.302545
(Iteration 48201 / 49000) loss: 2.302608
(Epoch 197 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 48301 / 49000) loss: 2.302628
(Iteration 48401 / 49000) loss: 2.302649
(Iteration 48501 / 49000) loss: 2.302836
(Epoch 198 / 200) train acc: 0.100000; val_acc: 0.078000
(Iteration 48601 / 49000) loss: 2.302417
(Iteration 48701 / 49000) loss: 2.302403
(Epoch 199 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 48801 / 49000) loss: 2.302477
(Iteration 48901 / 49000) loss: 2.302833
(Epoch 200 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 1 / 49000) loss: 530.948353
(Epoch 0 / 200) train acc: 0.103000; val_acc: 0.110000
(Iteration 101 / 49000) loss: 3.510640
(Iteration 201 / 49000) loss: 2.306161
(Epoch 1 / 200) train acc: 0.107000; val_acc: 0.102000
(Iteration 301 / 49000) loss: 2.302347
(Iteration 401 / 49000) loss: 2.302505
(Epoch 2 / 200) train acc: 0.111000; val_acc: 0.078000
```

```
(Iteration 501 / 49000) loss: 2.302847
(Iteration 601 / 49000) loss: 2.301489
(Iteration 701 / 49000) loss: 2.302120
(Epoch 3 / 200) train acc: 0.081000; val_acc: 0.078000
(Iteration 801 / 49000) loss: 2.302006
(Iteration 901 / 49000) loss: 2.303033
(Epoch 4 / 200) train acc: 0.101000; val_acc: 0.087000
(Iteration 1001 / 49000) loss: 2.302217
(Iteration 1101 / 49000) loss: 2.302302
(Iteration 1201 / 49000) loss: 2.303248
(Epoch 5 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 1301 / 49000) loss: 2.302812
(Iteration 1401 / 49000) loss: 2.303603
(Epoch 6 / 200) train acc: 0.084000; val_acc: 0.087000
(Iteration 1501 / 49000) loss: 2.302876
(Iteration 1601 / 49000) loss: 2.302200
(Iteration 1701 / 49000) loss: 2.301152
(Epoch 7 / 200) train acc: 0.093000; val_acc: 0.102000
(Iteration 1801 / 49000) loss: 2.302769
(Iteration 1901 / 49000) loss: 2.302418
(Epoch 8 / 200) train acc: 0.099000; val_acc: 0.112000
(Iteration 2001 / 49000) loss: 2.302771
(Iteration 2101 / 49000) loss: 2.302850
(Iteration 2201 / 49000) loss: 2.303068
(Epoch 9 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 2301 / 49000) loss: 2.302565
(Iteration 2401 / 49000) loss: 2.303576
(Epoch 10 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 2501 / 49000) loss: 2.302998
(Iteration 2601 / 49000) loss: 2.302520
(Epoch 11 / 200) train acc: 0.100000; val_acc: 0.113000
(Iteration 2701 / 49000) loss: 2.302960
(Iteration 2801 / 49000) loss: 2.303423
(Iteration 2901 / 49000) loss: 2.303588
(Epoch 12 / 200) train acc: 0.079000; val_acc: 0.078000
(Iteration 3001 / 49000) loss: 2.302155
(Iteration 3101 / 49000) loss: 2.303300
(Epoch 13 / 200) train acc: 0.098000; val_acc: 0.105000
(Iteration 3201 / 49000) loss: 2.303545
(Iteration 3301 / 49000) loss: 2.303187
(Iteration 3401 / 49000) loss: 2.302762
(Epoch 14 / 200) train acc: 0.101000; val_acc: 0.105000
(Iteration 3501 / 49000) loss: 2.302635
(Iteration 3601 / 49000) loss: 2.302430
(Epoch 15 / 200) train acc: 0.099000; val_acc: 0.105000
(Iteration 3701 / 49000) loss: 2.302882
(Iteration 3801 / 49000) loss: 2.303241
(Iteration 3901 / 49000) loss: 2.301648
```

```
(Epoch 16 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 4001 / 49000) loss: 2.302049
(Iteration 4101 / 49000) loss: 2.302039
(Epoch 17 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 4201 / 49000) loss: 2.301601
(Iteration 4301 / 49000) loss: 2.303172
(Iteration 4401 / 49000) loss: 2.302776
(Epoch 18 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 4501 / 49000) loss: 2.302785
(Iteration 4601 / 49000) loss: 2.301694
(Epoch 19 / 200) train acc: 0.119000; val_acc: 0.079000
(Iteration 4701 / 49000) loss: 2.303076
(Iteration 4801 / 49000) loss: 2.302467
(Epoch 20 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 4901 / 49000) loss: 2.302104
(Iteration 5001 / 49000) loss: 2.301947
(Iteration 5101 / 49000) loss: 2.302498
(Epoch 21 / 200) train acc: 0.088000; val_acc: 0.087000
(Iteration 5201 / 49000) loss: 2.302080
(Iteration 5301 / 49000) loss: 2.303295
(Epoch 22 / 200) train acc: 0.101000; val_acc: 0.113000
(Iteration 5401 / 49000) loss: 2.302704
(Iteration 5501 / 49000) loss: 2.302277
(Iteration 5601 / 49000) loss: 2.302789
(Epoch 23 / 200) train acc: 0.096000; val_acc: 0.105000
(Iteration 5701 / 49000) loss: 2.303139
(Iteration 5801 / 49000) loss: 2.302883
(Epoch 24 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 5901 / 49000) loss: 2.303062
(Iteration 6001 / 49000) loss: 2.302432
(Iteration 6101 / 49000) loss: 2.302535
(Epoch 25 / 200) train acc: 0.097000; val_acc: 0.119000
(Iteration 6201 / 49000) loss: 2.302767
(Iteration 6301 / 49000) loss: 2.302353
(Epoch 26 / 200) train acc: 0.094000; val_acc: 0.107000
(Iteration 6401 / 49000) loss: 2.303034
(Iteration 6501 / 49000) loss: 2.302702
(Iteration 6601 / 49000) loss: 2.302538
(Epoch 27 / 200) train acc: 0.126000; val_acc: 0.078000
(Iteration 6701 / 49000) loss: 2.303197
(Iteration 6801 / 49000) loss: 2.302676
(Epoch 28 / 200) train acc: 0.089000; val_acc: 0.078000
(Iteration 6901 / 49000) loss: 2.303009
(Iteration 7001 / 49000) loss: 2.301900
(Iteration 7101 / 49000) loss: 2.302669
(Epoch 29 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 7201 / 49000) loss: 2.302472
(Iteration 7301 / 49000) loss: 2.302639
```

```
(Epoch 30 / 200) train acc: 0.109000; val_acc: 0.078000
(Iteration 7401 / 49000) loss: 2.302694
(Iteration 7501 / 49000) loss: 2.302487
(Epoch 31 / 200) train acc: 0.099000; val_acc: 0.105000
(Iteration 7601 / 49000) loss: 2.301740
(Iteration 7701 / 49000) loss: 2.302447
(Iteration 7801 / 49000) loss: 2.301914
(Epoch 32 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 7901 / 49000) loss: 2.302162
(Iteration 8001 / 49000) loss: 2.302549
(Epoch 33 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 8101 / 49000) loss: 2.302065
(Iteration 8201 / 49000) loss: 2.303122
(Iteration 8301 / 49000) loss: 2.303223
(Epoch 34 / 200) train acc: 0.112000; val_acc: 0.079000
(Iteration 8401 / 49000) loss: 2.302408
(Iteration 8501 / 49000) loss: 2.302218
(Epoch 35 / 200) train acc: 0.084000; val_acc: 0.113000
(Iteration 8601 / 49000) loss: 2.302583
(Iteration 8701 / 49000) loss: 2.302524
(Iteration 8801 / 49000) loss: 2.302207
(Epoch 36 / 200) train acc: 0.112000; val_acc: 0.107000
(Iteration 8901 / 49000) loss: 2.302839
(Iteration 9001 / 49000) loss: 2.302486
(Epoch 37 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 9101 / 49000) loss: 2.302738
(Iteration 9201 / 49000) loss: 2.302799
(Iteration 9301 / 49000) loss: 2.302910
(Epoch 38 / 200) train acc: 0.113000; val_acc: 0.087000
(Iteration 9401 / 49000) loss: 2.303231
(Iteration 9501 / 49000) loss: 2.301863
(Epoch 39 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 9601 / 49000) loss: 2.302796
(Iteration 9701 / 49000) loss: 2.302514
(Epoch 40 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 9801 / 49000) loss: 2.301682
(Iteration 9901 / 49000) loss: 2.302208
(Iteration 10001 / 49000) loss: 2.301782
(Epoch 41 / 200) train acc: 0.083000; val_acc: 0.078000
(Iteration 10101 / 49000) loss: 2.302350
(Iteration 10201 / 49000) loss: 2.301981
(Epoch 42 / 200) train acc: 0.116000; val_acc: 0.079000
(Iteration 10301 / 49000) loss: 2.302580
(Iteration 10401 / 49000) loss: 2.302923
(Iteration 10501 / 49000) loss: 2.302025
(Epoch 43 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 10601 / 49000) loss: 2.302942
(Iteration 10701 / 49000) loss: 2.303434
```

```
(Epoch 44 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 10801 / 49000) loss: 2.302622
(Iteration 10901 / 49000) loss: 2.302893
(Iteration 11001 / 49000) loss: 2.303095
(Epoch 45 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 11101 / 49000) loss: 2.302281
(Iteration 11201 / 49000) loss: 2.302568
(Epoch 46 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 11301 / 49000) loss: 2.302448
(Iteration 11401 / 49000) loss: 2.302350
(Iteration 11501 / 49000) loss: 2.302203
(Epoch 47 / 200) train acc: 0.087000; val_acc: 0.079000
(Iteration 11601 / 49000) loss: 2.302499
(Iteration 11701 / 49000) loss: 2.303069
(Epoch 48 / 200) train acc: 0.104000; val_acc: 0.098000
(Iteration 11801 / 49000) loss: 2.302860
(Iteration 11901 / 49000) loss: 2.302212
(Iteration 12001 / 49000) loss: 2.302877
(Epoch 49 / 200) train acc: 0.106000; val_acc: 0.098000
(Iteration 12101 / 49000) loss: 2.302777
(Iteration 12201 / 49000) loss: 2.302840
(Epoch 50 / 200) train acc: 0.086000; val_acc: 0.079000
(Iteration 12301 / 49000) loss: 2.302481
(Iteration 12401 / 49000) loss: 2.302418
(Epoch 51 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 12501 / 49000) loss: 2.303029
(Iteration 12601 / 49000) loss: 2.301960
(Iteration 12701 / 49000) loss: 2.302861
(Epoch 52 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 12801 / 49000) loss: 2.302668
(Iteration 12901 / 49000) loss: 2.302865
(Epoch 53 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 13001 / 49000) loss: 2.302668
(Iteration 13101 / 49000) loss: 2.302150
(Iteration 13201 / 49000) loss: 2.302737
(Epoch 54 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 13301 / 49000) loss: 2.303249
(Iteration 13401 / 49000) loss: 2.302045
(Epoch 55 / 200) train acc: 0.095000; val_acc: 0.087000
(Iteration 13501 / 49000) loss: 2.302044
(Iteration 13601 / 49000) loss: 2.302934
(Iteration 13701 / 49000) loss: 2.301588
(Epoch 56 / 200) train acc: 0.113000; val_acc: 0.079000
(Iteration 13801 / 49000) loss: 2.303364
(Iteration 13901 / 49000) loss: 2.302326
(Epoch 57 / 200) train acc: 0.105000; val_acc: 0.102000
(Iteration 14001 / 49000) loss: 2.302801
(Iteration 14101 / 49000) loss: 2.302116
```

```
(Iteration 14201 / 49000) loss: 2.302937
(Epoch 58 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 14301 / 49000) loss: 2.302803
(Iteration 14401 / 49000) loss: 2.302497
(Epoch 59 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 14501 / 49000) loss: 2.302112
(Iteration 14601 / 49000) loss: 2.302977
(Epoch 60 / 200) train acc: 0.112000; val_acc: 0.105000
(Iteration 14701 / 49000) loss: 2.303632
(Iteration 14801 / 49000) loss: 2.302119
(Iteration 14901 / 49000) loss: 2.302260
(Epoch 61 / 200) train acc: 0.098000; val_acc: 0.098000
(Iteration 15001 / 49000) loss: 2.302398
(Iteration 15101 / 49000) loss: 2.302424
(Epoch 62 / 200) train acc: 0.112000; val_acc: 0.098000
(Iteration 15201 / 49000) loss: 2.303081
(Iteration 15301 / 49000) loss: 2.303219
(Iteration 15401 / 49000) loss: 2.302804
(Epoch 63 / 200) train acc: 0.084000; val_acc: 0.098000
(Iteration 15501 / 49000) loss: 2.302688
(Iteration 15601 / 49000) loss: 2.302968
(Epoch 64 / 200) train acc: 0.112000; val_acc: 0.098000
(Iteration 15701 / 49000) loss: 2.302730
(Iteration 15801 / 49000) loss: 2.302362
(Iteration 15901 / 49000) loss: 2.301922
(Epoch 65 / 200) train acc: 0.104000; val_acc: 0.098000
(Iteration 16001 / 49000) loss: 2.302710
(Iteration 16101 / 49000) loss: 2.302379
(Epoch 66 / 200) train acc: 0.092000; val_acc: 0.098000
(Iteration 16201 / 49000) loss: 2.302035
(Iteration 16301 / 49000) loss: 2.302278
(Iteration 16401 / 49000) loss: 2.302835
(Epoch 67 / 200) train acc: 0.111000; val_acc: 0.098000
(Iteration 16501 / 49000) loss: 2.302453
(Iteration 16601 / 49000) loss: 2.302908
(Epoch 68 / 200) train acc: 0.111000; val_acc: 0.079000
(Iteration 16701 / 49000) loss: 2.301921
(Iteration 16801 / 49000) loss: 2.303313
(Iteration 16901 / 49000) loss: 2.301961
(Epoch 69 / 200) train acc: 0.118000; val_acc: 0.079000
(Iteration 17001 / 49000) loss: 2.301979
(Iteration 17101 / 49000) loss: 2.302278
(Epoch 70 / 200) train acc: 0.089000; val_acc: 0.098000
(Iteration 17201 / 49000) loss: 2.302291
(Iteration 17301 / 49000) loss: 2.302164
(Epoch 71 / 200) train acc: 0.101000; val_acc: 0.098000
(Iteration 17401 / 49000) loss: 2.303048
(Iteration 17501 / 49000) loss: 2.303077
```

```
(Iteration 17601 / 49000) loss: 2.302188
(Epoch 72 / 200) train acc: 0.093000; val_acc: 0.098000
(Iteration 17701 / 49000) loss: 2.302534
(Iteration 17801 / 49000) loss: 2.302089
(Epoch 73 / 200) train acc: 0.101000; val_acc: 0.098000
(Iteration 17901 / 49000) loss: 2.302378
(Iteration 18001 / 49000) loss: 2.302969
(Iteration 18101 / 49000) loss: 2.302371
(Epoch 74 / 200) train acc: 0.090000; val_acc: 0.098000
(Iteration 18201 / 49000) loss: 2.302634
(Iteration 18301 / 49000) loss: 2.302661
(Epoch 75 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 18401 / 49000) loss: 2.302550
(Iteration 18501 / 49000) loss: 2.302431
(Iteration 18601 / 49000) loss: 2.302405
(Epoch 76 / 200) train acc: 0.113000; val_acc: 0.079000
(Iteration 18701 / 49000) loss: 2.302888
(Iteration 18801 / 49000) loss: 2.302653
(Epoch 77 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 18901 / 49000) loss: 2.302796
(Iteration 19001 / 49000) loss: 2.302596
(Iteration 19101 / 49000) loss: 2.302309
(Epoch 78 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 19201 / 49000) loss: 2.302141
(Iteration 19301 / 49000) loss: 2.302273
(Epoch 79 / 200) train acc: 0.112000; val_acc: 0.102000
(Iteration 19401 / 49000) loss: 2.302884
(Iteration 19501 / 49000) loss: 2.302536
(Epoch 80 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 19601 / 49000) loss: 2.303037
(Iteration 19701 / 49000) loss: 2.302994
(Iteration 19801 / 49000) loss: 2.302612
(Epoch 81 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 19901 / 49000) loss: 2.302364
(Iteration 20001 / 49000) loss: 2.302574
(Epoch 82 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 20101 / 49000) loss: 2.302430
(Iteration 20201 / 49000) loss: 2.302381
(Iteration 20301 / 49000) loss: 2.302710
(Epoch 83 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 20401 / 49000) loss: 2.302827
(Iteration 20501 / 49000) loss: 2.302234
(Epoch 84 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 20601 / 49000) loss: 2.302247
(Iteration 20701 / 49000) loss: 2.302793
(Iteration 20801 / 49000) loss: 2.302945
(Epoch 85 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 20901 / 49000) loss: 2.302710
```

```
(Iteration 21001 / 49000) loss: 2.302961
(Epoch 86 / 200) train acc: 0.124000; val_acc: 0.102000
(Iteration 21101 / 49000) loss: 2.302744
(Iteration 21201 / 49000) loss: 2.302351
(Iteration 21301 / 49000) loss: 2.302387
(Epoch 87 / 200) train acc: 0.116000; val_acc: 0.079000
(Iteration 21401 / 49000) loss: 2.302969
(Iteration 21501 / 49000) loss: 2.302373
(Epoch 88 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 21601 / 49000) loss: 2.302552
(Iteration 21701 / 49000) loss: 2.302536
(Iteration 21801 / 49000) loss: 2.302830
(Epoch 89 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 21901 / 49000) loss: 2.302981
(Iteration 22001 / 49000) loss: 2.302846
(Epoch 90 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 22101 / 49000) loss: 2.302551
(Iteration 22201 / 49000) loss: 2.302701
(Epoch 91 / 200) train acc: 0.116000; val_acc: 0.079000
(Iteration 22301 / 49000) loss: 2.302607
(Iteration 22401 / 49000) loss: 2.302784
(Iteration 22501 / 49000) loss: 2.302876
(Epoch 92 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 22601 / 49000) loss: 2.302664
(Iteration 22701 / 49000) loss: 2.302662
(Epoch 93 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 22801 / 49000) loss: 2.302570
(Iteration 22901 / 49000) loss: 2.302201
(Iteration 23001 / 49000) loss: 2.302714
(Epoch 94 / 200) train acc: 0.110000; val_acc: 0.079000
(Iteration 23101 / 49000) loss: 2.302597
(Iteration 23201 / 49000) loss: 2.302799
(Epoch 95 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 23301 / 49000) loss: 2.302475
(Iteration 23401 / 49000) loss: 2.302542
(Iteration 23501 / 49000) loss: 2.302467
(Epoch 96 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 23601 / 49000) loss: 2.302537
(Iteration 23701 / 49000) loss: 2.302666
(Epoch 97 / 200) train acc: 0.100000; val_acc: 0.078000
(Iteration 23801 / 49000) loss: 2.302750
(Iteration 23901 / 49000) loss: 2.301926
(Iteration 24001 / 49000) loss: 2.303136
(Epoch 98 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 24101 / 49000) loss: 2.302225
(Iteration 24201 / 49000) loss: 2.302769
(Epoch 99 / 200) train acc: 0.091000; val_acc: 0.078000
(Iteration 24301 / 49000) loss: 2.302893
```

```
(Iteration 24401 / 49000) loss: 2.302050
(Epoch 100 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 24501 / 49000) loss: 2.302893
(Iteration 24601 / 49000) loss: 2.302619
(Iteration 24701 / 49000) loss: 2.302821
(Epoch 101 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 24801 / 49000) loss: 2.302134
(Iteration 24901 / 49000) loss: 2.302302
(Epoch 102 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 25001 / 49000) loss: 2.302799
(Iteration 25101 / 49000) loss: 2.302719
(Iteration 25201 / 49000) loss: 2.302469
(Epoch 103 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 25301 / 49000) loss: 2.302267
(Iteration 25401 / 49000) loss: 2.302412
(Epoch 104 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 25501 / 49000) loss: 2.302605
(Iteration 25601 / 49000) loss: 2.302569
(Iteration 25701 / 49000) loss: 2.302056
(Epoch 105 / 200) train acc: 0.091000; val_acc: 0.078000
(Iteration 25801 / 49000) loss: 2.302732
(Iteration 25901 / 49000) loss: 2.302812
(Epoch 106 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 26001 / 49000) loss: 2.302544
(Iteration 26101 / 49000) loss: 2.302553
(Iteration 26201 / 49000) loss: 2.302753
(Epoch 107 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 26301 / 49000) loss: 2.302865
(Iteration 26401 / 49000) loss: 2.302462
(Epoch 108 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 26501 / 49000) loss: 2.302824
(Iteration 26601 / 49000) loss: 2.302102
(Iteration 26701 / 49000) loss: 2.302461
(Epoch 109 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 26801 / 49000) loss: 2.303018
(Iteration 26901 / 49000) loss: 2.302328
(Epoch 110 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 27001 / 49000) loss: 2.302592
(Iteration 27101 / 49000) loss: 2.302889
(Epoch 111 / 200) train acc: 0.095000; val_acc: 0.078000
(Iteration 27201 / 49000) loss: 2.302259
(Iteration 27301 / 49000) loss: 2.303095
(Iteration 27401 / 49000) loss: 2.301670
(Epoch 112 / 200) train acc: 0.112000; val_acc: 0.079000
(Iteration 27501 / 49000) loss: 2.302503
(Iteration 27601 / 49000) loss: 2.302007
(Epoch 113 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 27701 / 49000) loss: 2.302807
```

```
(Iteration 27801 / 49000) loss: 2.302374
(Iteration 27901 / 49000) loss: 2.301750
(Epoch 114 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 28001 / 49000) loss: 2.302681
(Iteration 28101 / 49000) loss: 2.302488
(Epoch 115 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 28201 / 49000) loss: 2.302741
(Iteration 28301 / 49000) loss: 2.302498
(Iteration 28401 / 49000) loss: 2.303069
(Epoch 116 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 28501 / 49000) loss: 2.302838
(Iteration 28601 / 49000) loss: 2.302416
(Epoch 117 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 28701 / 49000) loss: 2.302745
(Iteration 28801 / 49000) loss: 2.301769
(Iteration 28901 / 49000) loss: 2.302589
(Epoch 118 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 29001 / 49000) loss: 2.301872
(Iteration 29101 / 49000) loss: 2.302418
(Epoch 119 / 200) train acc: 0.100000; val_acc: 0.078000
(Iteration 29201 / 49000) loss: 2.302732
(Iteration 29301 / 49000) loss: 2.302855
(Epoch 120 / 200) train acc: 0.103000; val_acc: 0.078000
(Iteration 29401 / 49000) loss: 2.301967
(Iteration 29501 / 49000) loss: 2.302684
(Iteration 29601 / 49000) loss: 2.302243
(Epoch 121 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 29701 / 49000) loss: 2.302736
(Iteration 29801 / 49000) loss: 2.302229
(Epoch 122 / 200) train acc: 0.087000; val_acc: 0.078000
(Iteration 29901 / 49000) loss: 2.302307
(Iteration 30001 / 49000) loss: 2.301961
(Iteration 30101 / 49000) loss: 2.303278
(Epoch 123 / 200) train acc: 0.087000; val_acc: 0.078000
(Iteration 30201 / 49000) loss: 2.302340
(Iteration 30301 / 49000) loss: 2.303557
(Epoch 124 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 30401 / 49000) loss: 2.302848
(Iteration 30501 / 49000) loss: 2.302236
(Iteration 30601 / 49000) loss: 2.302743
(Epoch 125 / 200) train acc: 0.113000; val_acc: 0.078000
(Iteration 30701 / 49000) loss: 2.302374
(Iteration 30801 / 49000) loss: 2.302366
(Epoch 126 / 200) train acc: 0.092000; val_acc: 0.079000
(Iteration 30901 / 49000) loss: 2.303274
(Iteration 31001 / 49000) loss: 2.302315
(Iteration 31101 / 49000) loss: 2.302591
(Epoch 127 / 200) train acc: 0.089000; val_acc: 0.079000
```

```
(Iteration 31201 / 49000) loss: 2.303147
(Iteration 31301 / 49000) loss: 2.302560
(Epoch 128 / 200) train acc: 0.086000; val_acc: 0.079000
(Iteration 31401 / 49000) loss: 2.302809
(Iteration 31501 / 49000) loss: 2.302457
(Iteration 31601 / 49000) loss: 2.301895
(Epoch 129 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 31701 / 49000) loss: 2.301872
(Iteration 31801 / 49000) loss: 2.302429
(Epoch 130 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 31901 / 49000) loss: 2.302726
(Iteration 32001 / 49000) loss: 2.302698
(Epoch 131 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 32101 / 49000) loss: 2.301899
(Iteration 32201 / 49000) loss: 2.303137
(Iteration 32301 / 49000) loss: 2.303077
(Epoch 132 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 32401 / 49000) loss: 2.302163
(Iteration 32501 / 49000) loss: 2.302246
(Epoch 133 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 32601 / 49000) loss: 2.302090
(Iteration 32701 / 49000) loss: 2.301828
(Iteration 32801 / 49000) loss: 2.302448
(Epoch 134 / 200) train acc: 0.115000; val_acc: 0.079000
(Iteration 32901 / 49000) loss: 2.302560
(Iteration 33001 / 49000) loss: 2.302779
(Epoch 135 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 33101 / 49000) loss: 2.302196
(Iteration 33201 / 49000) loss: 2.302552
(Iteration 33301 / 49000) loss: 2.302484
(Epoch 136 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 33401 / 49000) loss: 2.302354
(Iteration 33501 / 49000) loss: 2.302464
(Epoch 137 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 33601 / 49000) loss: 2.302317
(Iteration 33701 / 49000) loss: 2.302798
(Iteration 33801 / 49000) loss: 2.302395
(Epoch 138 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 33901 / 49000) loss: 2.302952
(Iteration 34001 / 49000) loss: 2.302912
(Epoch 139 / 200) train acc: 0.114000; val_acc: 0.078000
(Iteration 34101 / 49000) loss: 2.302671
(Iteration 34201 / 49000) loss: 2.302794
(Epoch 140 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 34301 / 49000) loss: 2.302027
(Iteration 34401 / 49000) loss: 2.303086
(Iteration 34501 / 49000) loss: 2.303033
(Epoch 141 / 200) train acc: 0.097000; val_acc: 0.079000
```

```
(Iteration 34601 / 49000) loss: 2.302310
(Iteration 34701 / 49000) loss: 2.302616
(Epoch 142 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 34801 / 49000) loss: 2.303057
(Iteration 34901 / 49000) loss: 2.302109
(Iteration 35001 / 49000) loss: 2.303185
(Epoch 143 / 200) train acc: 0.097000; val_acc: 0.079000
(Iteration 35101 / 49000) loss: 2.302141
(Iteration 35201 / 49000) loss: 2.301709
(Epoch 144 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 35301 / 49000) loss: 2.303144
(Iteration 35401 / 49000) loss: 2.302728
(Iteration 35501 / 49000) loss: 2.302431
(Epoch 145 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 35601 / 49000) loss: 2.302698
(Iteration 35701 / 49000) loss: 2.302662
(Epoch 146 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 35801 / 49000) loss: 2.302465
(Iteration 35901 / 49000) loss: 2.303011
(Iteration 36001 / 49000) loss: 2.302679
(Epoch 147 / 200) train acc: 0.118000; val_acc: 0.078000
(Iteration 36101 / 49000) loss: 2.302832
(Iteration 36201 / 49000) loss: 2.302619
(Epoch 148 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 36301 / 49000) loss: 2.302688
(Iteration 36401 / 49000) loss: 2.302691
(Iteration 36501 / 49000) loss: 2.302621
(Epoch 149 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 36601 / 49000) loss: 2.302610
(Iteration 36701 / 49000) loss: 2.302642
(Epoch 150 / 200) train acc: 0.112000; val_acc: 0.079000
(Iteration 36801 / 49000) loss: 2.302797
(Iteration 36901 / 49000) loss: 2.302482
(Epoch 151 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 37001 / 49000) loss: 2.302400
(Iteration 37101 / 49000) loss: 2.302295
(Iteration 37201 / 49000) loss: 2.302707
(Epoch 152 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 37301 / 49000) loss: 2.302363
(Iteration 37401 / 49000) loss: 2.302233
(Epoch 153 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 37501 / 49000) loss: 2.302292
(Iteration 37601 / 49000) loss: 2.302669
(Iteration 37701 / 49000) loss: 2.302118
(Epoch 154 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 37801 / 49000) loss: 2.302712
(Iteration 37901 / 49000) loss: 2.301979
(Epoch 155 / 200) train acc: 0.105000; val_acc: 0.079000
```

```
(Iteration 38001 / 49000) loss: 2.302372
(Iteration 38101 / 49000) loss: 2.302511
(Iteration 38201 / 49000) loss: 2.302213
(Epoch 156 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 38301 / 49000) loss: 2.302686
(Iteration 38401 / 49000) loss: 2.302549
(Epoch 157 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 38501 / 49000) loss: 2.302542
(Iteration 38601 / 49000) loss: 2.302946
(Iteration 38701 / 49000) loss: 2.302661
(Epoch 158 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 38801 / 49000) loss: 2.302617
(Iteration 38901 / 49000) loss: 2.302368
(Epoch 159 / 200) train acc: 0.113000; val_acc: 0.079000
(Iteration 39001 / 49000) loss: 2.302518
(Iteration 39101 / 49000) loss: 2.302201
(Epoch 160 / 200) train acc: 0.083000; val_acc: 0.079000
(Iteration 39201 / 49000) loss: 2.302729
(Iteration 39301 / 49000) loss: 2.302375
(Iteration 39401 / 49000) loss: 2.302776
(Epoch 161 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 39501 / 49000) loss: 2.302903
(Iteration 39601 / 49000) loss: 2.302735
(Epoch 162 / 200) train acc: 0.092000; val_acc: 0.079000
(Iteration 39701 / 49000) loss: 2.302284
(Iteration 39801 / 49000) loss: 2.302815
(Iteration 39901 / 49000) loss: 2.302485
(Epoch 163 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 40001 / 49000) loss: 2.302690
(Iteration 40101 / 49000) loss: 2.302664
(Epoch 164 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 40201 / 49000) loss: 2.302395
(Iteration 40301 / 49000) loss: 2.302641
(Iteration 40401 / 49000) loss: 2.302676
(Epoch 165 / 200) train acc: 0.103000; val_acc: 0.087000
(Iteration 40501 / 49000) loss: 2.302678
(Iteration 40601 / 49000) loss: 2.302942
(Epoch 166 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 40701 / 49000) loss: 2.302734
(Iteration 40801 / 49000) loss: 2.302357
(Iteration 40901 / 49000) loss: 2.302674
(Epoch 167 / 200) train acc: 0.083000; val_acc: 0.087000
(Iteration 41001 / 49000) loss: 2.302435
(Iteration 41101 / 49000) loss: 2.302722
(Epoch 168 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 41201 / 49000) loss: 2.302631
(Iteration 41301 / 49000) loss: 2.302749
(Iteration 41401 / 49000) loss: 2.302397
```

```
(Epoch 169 / 200) train acc: 0.107000; val_acc: 0.087000
(Iteration 41501 / 49000) loss: 2.302730
(Iteration 41601 / 49000) loss: 2.302984
(Epoch 170 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 41701 / 49000) loss: 2.302754
(Iteration 41801 / 49000) loss: 2.302581
(Epoch 171 / 200) train acc: 0.100000; val_acc: 0.087000
(Iteration 41901 / 49000) loss: 2.302878
(Iteration 42001 / 49000) loss: 2.302422
(Iteration 42101 / 49000) loss: 2.302568
(Epoch 172 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 42201 / 49000) loss: 2.302593
(Iteration 42301 / 49000) loss: 2.303120
(Epoch 173 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 42401 / 49000) loss: 2.302693
(Iteration 42501 / 49000) loss: 2.302563
(Iteration 42601 / 49000) loss: 2.302917
(Epoch 174 / 200) train acc: 0.106000; val_acc: 0.087000
(Iteration 42701 / 49000) loss: 2.302680
(Iteration 42801 / 49000) loss: 2.302693
(Epoch 175 / 200) train acc: 0.106000; val_acc: 0.087000
(Iteration 42901 / 49000) loss: 2.302482
(Iteration 43001 / 49000) loss: 2.302659
(Iteration 43101 / 49000) loss: 2.302665
(Epoch 176 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 43201 / 49000) loss: 2.302565
(Iteration 43301 / 49000) loss: 2.302493
(Epoch 177 / 200) train acc: 0.097000; val_acc: 0.087000
(Iteration 43401 / 49000) loss: 2.302568
(Iteration 43501 / 49000) loss: 2.302775
(Iteration 43601 / 49000) loss: 2.302374
(Epoch 178 / 200) train acc: 0.107000; val_acc: 0.087000
(Iteration 43701 / 49000) loss: 2.302634
(Iteration 43801 / 49000) loss: 2.303131
(Epoch 179 / 200) train acc: 0.106000; val_acc: 0.087000
(Iteration 43901 / 49000) loss: 2.302721
(Iteration 44001 / 49000) loss: 2.302630
(Epoch 180 / 200) train acc: 0.087000; val_acc: 0.087000
(Iteration 44101 / 49000) loss: 2.302835
(Iteration 44201 / 49000) loss: 2.302871
(Iteration 44301 / 49000) loss: 2.302316
(Epoch 181 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 44401 / 49000) loss: 2.302348
(Iteration 44501 / 49000) loss: 2.302431
(Epoch 182 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 44601 / 49000) loss: 2.302640
(Iteration 44701 / 49000) loss: 2.302372
(Iteration 44801 / 49000) loss: 2.302785
```

```
(Epoch 183 / 200) train acc: 0.099000; val_acc: 0.078000
(Iteration 44901 / 49000) loss: 2.302953
(Iteration 45001 / 49000) loss: 2.302419
(Epoch 184 / 200) train acc: 0.111000; val_acc: 0.078000
(Iteration 45101 / 49000) loss: 2.302620
(Iteration 45201 / 49000) loss: 2.302737
(Iteration 45301 / 49000) loss: 2.302580
(Epoch 185 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 45401 / 49000) loss: 2.302500
(Iteration 45501 / 49000) loss: 2.302337
(Epoch 186 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 45601 / 49000) loss: 2.302446
(Iteration 45701 / 49000) loss: 2.302294
(Iteration 45801 / 49000) loss: 2.302621
(Epoch 187 / 200) train acc: 0.091000; val_acc: 0.078000
(Iteration 45901 / 49000) loss: 2.302520
(Iteration 46001 / 49000) loss: 2.302309
(Epoch 188 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 46101 / 49000) loss: 2.302522
(Iteration 46201 / 49000) loss: 2.302530
(Iteration 46301 / 49000) loss: 2.302927
(Epoch 189 / 200) train acc: 0.082000; val_acc: 0.078000
(Iteration 46401 / 49000) loss: 2.302481
(Iteration 46501 / 49000) loss: 2.302386
(Epoch 190 / 200) train acc: 0.091000; val_acc: 0.078000
(Iteration 46601 / 49000) loss: 2.302900
(Iteration 46701 / 49000) loss: 2.302337
(Epoch 191 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 46801 / 49000) loss: 2.302749
(Iteration 46901 / 49000) loss: 2.302335
(Iteration 47001 / 49000) loss: 2.302473
(Epoch 192 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 47101 / 49000) loss: 2.302577
(Iteration 47201 / 49000) loss: 2.302502
(Epoch 193 / 200) train acc: 0.091000; val_acc: 0.078000
(Iteration 47301 / 49000) loss: 2.302755
(Iteration 47401 / 49000) loss: 2.302534
(Iteration 47501 / 49000) loss: 2.302328
(Epoch 194 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 47601 / 49000) loss: 2.302844
(Iteration 47701 / 49000) loss: 2.302459
(Epoch 195 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 47801 / 49000) loss: 2.302320
(Iteration 47901 / 49000) loss: 2.302510
(Iteration 48001 / 49000) loss: 2.302627
(Epoch 196 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 48101 / 49000) loss: 2.302795
(Iteration 48201 / 49000) loss: 2.302584
```

```
(Epoch 197 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 48301 / 49000) loss: 2.302542
(Iteration 48401 / 49000) loss: 2.302710
(Iteration 48501 / 49000) loss: 2.302842
(Epoch 198 / 200) train acc: 0.115000; val_acc: 0.079000
(Iteration 48601 / 49000) loss: 2.302864
(Iteration 48701 / 49000) loss: 2.302665
(Epoch 199 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 48801 / 49000) loss: 2.302507
(Iteration 48901 / 49000) loss: 2.302332
(Epoch 200 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 1 / 49000) loss: 522.171232
(Epoch 0 / 200) train acc: 0.103000; val_acc: 0.109000
(Iteration 101 / 49000) loss: 3.648542
(Iteration 201 / 49000) loss: 2.306595
(Epoch 1 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 301 / 49000) loss: 2.303772
(Iteration 401 / 49000) loss: 2.302418
(Epoch 2 / 200) train acc: 0.116000; val_acc: 0.078000
(Iteration 501 / 49000) loss: 2.302388
(Iteration 601 / 49000) loss: 2.303604
(Iteration 701 / 49000) loss: 2.302855
(Epoch 3 / 200) train acc: 0.095000; val_acc: 0.078000
(Iteration 801 / 49000) loss: 2.302756
(Iteration 901 / 49000) loss: 2.303444
(Epoch 4 / 200) train acc: 0.113000; val_acc: 0.087000
(Iteration 1001 / 49000) loss: 2.301508
(Iteration 1101 / 49000) loss: 2.302758
(Iteration 1201 / 49000) loss: 2.303040
(Epoch 5 / 200) train acc: 0.087000; val_acc: 0.087000
(Iteration 1301 / 49000) loss: 2.301929
(Iteration 1401 / 49000) loss: 2.302357
(Epoch 6 / 200) train acc: 0.094000; val_acc: 0.105000
(Iteration 1501 / 49000) loss: 2.302972
(Iteration 1601 / 49000) loss: 2.303257
(Iteration 1701 / 49000) loss: 2.303433
(Epoch 7 / 200) train acc: 0.100000; val_acc: 0.105000
(Iteration 1801 / 49000) loss: 2.302061
(Iteration 1901 / 49000) loss: 2.302294
(Epoch 8 / 200) train acc: 0.107000; val_acc: 0.105000
(Iteration 2001 / 49000) loss: 2.302398
(Iteration 2101 / 49000) loss: 2.301613
(Iteration 2201 / 49000) loss: 2.302822
(Epoch 9 / 200) train acc: 0.115000; val_acc: 0.078000
(Iteration 2301 / 49000) loss: 2.301930
(Iteration 2401 / 49000) loss: 2.303630
(Epoch 10 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 2501 / 49000) loss: 2.302494
```

```
(Iteration 2601 / 49000) loss: 2.302958
(Epoch 11 / 200) train acc: 0.099000; val_acc: 0.078000
(Iteration 2701 / 49000) loss: 2.301900
(Iteration 2801 / 49000) loss: 2.302450
(Iteration 2901 / 49000) loss: 2.303217
(Epoch 12 / 200) train acc: 0.076000; val_acc: 0.105000
(Iteration 3001 / 49000) loss: 2.302356
(Iteration 3101 / 49000) loss: 2.303226
(Epoch 13 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 3201 / 49000) loss: 2.302270
(Iteration 3301 / 49000) loss: 2.302455
(Iteration 3401 / 49000) loss: 2.302584
(Epoch 14 / 200) train acc: 0.081000; val_acc: 0.078000
(Iteration 3501 / 49000) loss: 2.303371
(Iteration 3601 / 49000) loss: 2.302716
(Epoch 15 / 200) train acc: 0.105000; val_acc: 0.102000
(Iteration 3701 / 49000) loss: 2.302984
(Iteration 3801 / 49000) loss: 2.302385
(Iteration 3901 / 49000) loss: 2.303156
(Epoch 16 / 200) train acc: 0.116000; val_acc: 0.078000
(Iteration 4001 / 49000) loss: 2.302576
(Iteration 4101 / 49000) loss: 2.302404
(Epoch 17 / 200) train acc: 0.111000; val_acc: 0.087000
(Iteration 4201 / 49000) loss: 2.302108
(Iteration 4301 / 49000) loss: 2.302805
(Iteration 4401 / 49000) loss: 2.303221
(Epoch 18 / 200) train acc: 0.105000; val_acc: 0.087000
(Iteration 4501 / 49000) loss: 2.302669
(Iteration 4601 / 49000) loss: 2.303464
(Epoch 19 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 4701 / 49000) loss: 2.301775
(Iteration 4801 / 49000) loss: 2.302039
(Epoch 20 / 200) train acc: 0.099000; val_acc: 0.105000
(Iteration 4901 / 49000) loss: 2.302932
(Iteration 5001 / 49000) loss: 2.302398
(Iteration 5101 / 49000) loss: 2.302343
(Epoch 21 / 200) train acc: 0.104000; val_acc: 0.105000
(Iteration 5201 / 49000) loss: 2.303047
(Iteration 5301 / 49000) loss: 2.302390
(Epoch 22 / 200) train acc: 0.082000; val_acc: 0.105000
(Iteration 5401 / 49000) loss: 2.302443
(Iteration 5501 / 49000) loss: 2.302500
(Iteration 5601 / 49000) loss: 2.302478
(Epoch 23 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 5701 / 49000) loss: 2.302181
(Iteration 5801 / 49000) loss: 2.303049
(Epoch 24 / 200) train acc: 0.090000; val_acc: 0.087000
(Iteration 5901 / 49000) loss: 2.303176
```

```
(Iteration 6001 / 49000) loss: 2.303216
(Iteration 6101 / 49000) loss: 2.302714
(Epoch 25 / 200) train acc: 0.089000; val_acc: 0.087000
(Iteration 6201 / 49000) loss: 2.302544
(Iteration 6301 / 49000) loss: 2.302541
(Epoch 26 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 6401 / 49000) loss: 2.302433
(Iteration 6501 / 49000) loss: 2.302703
(Iteration 6601 / 49000) loss: 2.303029
(Epoch 27 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 6701 / 49000) loss: 2.302812
(Iteration 6801 / 49000) loss: 2.302570
(Epoch 28 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 6901 / 49000) loss: 2.301894
(Iteration 7001 / 49000) loss: 2.303136
(Iteration 7101 / 49000) loss: 2.301956
(Epoch 29 / 200) train acc: 0.089000; val_acc: 0.078000
(Iteration 7201 / 49000) loss: 2.302797
(Iteration 7301 / 49000) loss: 2.302868
(Epoch 30 / 200) train acc: 0.092000; val_acc: 0.078000
(Iteration 7401 / 49000) loss: 2.302935
(Iteration 7501 / 49000) loss: 2.302940
(Epoch 31 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 7601 / 49000) loss: 2.301855
(Iteration 7701 / 49000) loss: 2.303315
(Iteration 7801 / 49000) loss: 2.303274
(Epoch 32 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 7901 / 49000) loss: 2.301796
(Iteration 8001 / 49000) loss: 2.303301
(Epoch 33 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 8101 / 49000) loss: 2.302453
(Iteration 8201 / 49000) loss: 2.302268
(Iteration 8301 / 49000) loss: 2.301744
(Epoch 34 / 200) train acc: 0.113000; val_acc: 0.079000
(Iteration 8401 / 49000) loss: 2.302954
(Iteration 8501 / 49000) loss: 2.303157
(Epoch 35 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 8601 / 49000) loss: 2.302717
(Iteration 8701 / 49000) loss: 2.302773
(Iteration 8801 / 49000) loss: 2.302646
(Epoch 36 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 8901 / 49000) loss: 2.302273
(Iteration 9001 / 49000) loss: 2.301998
(Epoch 37 / 200) train acc: 0.092000; val_acc: 0.078000
(Iteration 9101 / 49000) loss: 2.303297
(Iteration 9201 / 49000) loss: 2.302792
(Iteration 9301 / 49000) loss: 2.303265
(Epoch 38 / 200) train acc: 0.095000; val_acc: 0.078000
```

```
(Iteration 9401 / 49000) loss: 2.303327
(Iteration 9501 / 49000) loss: 2.301220
(Epoch 39 / 200) train acc: 0.110000; val_acc: 0.098000
(Iteration 9601 / 49000) loss: 2.302920
(Iteration 9701 / 49000) loss: 2.301306
(Epoch 40 / 200) train acc: 0.080000; val_acc: 0.098000
(Iteration 9801 / 49000) loss: 2.303114
(Iteration 9901 / 49000) loss: 2.303218
(Iteration 10001 / 49000) loss: 2.302621
(Epoch 41 / 200) train acc: 0.093000; val_acc: 0.098000
(Iteration 10101 / 49000) loss: 2.302420
(Iteration 10201 / 49000) loss: 2.302156
(Epoch 42 / 200) train acc: 0.095000; val_acc: 0.098000
(Iteration 10301 / 49000) loss: 2.302684
(Iteration 10401 / 49000) loss: 2.302275
(Iteration 10501 / 49000) loss: 2.302767
(Epoch 43 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 10601 / 49000) loss: 2.301910
(Iteration 10701 / 49000) loss: 2.303659
(Epoch 44 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 10801 / 49000) loss: 2.303127
(Iteration 10901 / 49000) loss: 2.302851
(Iteration 11001 / 49000) loss: 2.302802
(Epoch 45 / 200) train acc: 0.089000; val_acc: 0.078000
(Iteration 11101 / 49000) loss: 2.302544
(Iteration 11201 / 49000) loss: 2.302364
(Epoch 46 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 11301 / 49000) loss: 2.302793
(Iteration 11401 / 49000) loss: 2.303105
(Iteration 11501 / 49000) loss: 2.302840
(Epoch 47 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 11601 / 49000) loss: 2.301979
(Iteration 11701 / 49000) loss: 2.302601
(Epoch 48 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 11801 / 49000) loss: 2.302524
(Iteration 11901 / 49000) loss: 2.302438
(Iteration 12001 / 49000) loss: 2.302265
(Epoch 49 / 200) train acc: 0.111000; val_acc: 0.078000
(Iteration 12101 / 49000) loss: 2.302270
(Iteration 12201 / 49000) loss: 2.303040
(Epoch 50 / 200) train acc: 0.108000; val_acc: 0.078000
(Iteration 12301 / 49000) loss: 2.302274
(Iteration 12401 / 49000) loss: 2.302349
(Epoch 51 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 12501 / 49000) loss: 2.302910
(Iteration 12601 / 49000) loss: 2.302246
(Iteration 12701 / 49000) loss: 2.302289
(Epoch 52 / 200) train acc: 0.102000; val_acc: 0.078000
```

```
(Iteration 12801 / 49000) loss: 2.302278
(Iteration 12901 / 49000) loss: 2.302165
(Epoch 53 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 13001 / 49000) loss: 2.302637
(Iteration 13101 / 49000) loss: 2.302672
(Iteration 13201 / 49000) loss: 2.302649
(Epoch 54 / 200) train acc: 0.085000; val_acc: 0.113000
(Iteration 13301 / 49000) loss: 2.302977
(Iteration 13401 / 49000) loss: 2.302696
(Epoch 55 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 13501 / 49000) loss: 2.302451
(Iteration 13601 / 49000) loss: 2.302668
(Iteration 13701 / 49000) loss: 2.303126
(Epoch 56 / 200) train acc: 0.109000; val_acc: 0.078000
(Iteration 13801 / 49000) loss: 2.302555
(Iteration 13901 / 49000) loss: 2.303065
(Epoch 57 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 14001 / 49000) loss: 2.302559
(Iteration 14101 / 49000) loss: 2.302532
(Iteration 14201 / 49000) loss: 2.302532
(Epoch 58 / 200) train acc: 0.094000; val_acc: 0.119000
(Iteration 14301 / 49000) loss: 2.302602
(Iteration 14401 / 49000) loss: 2.302476
(Epoch 59 / 200) train acc: 0.100000; val_acc: 0.113000
(Iteration 14501 / 49000) loss: 2.302716
(Iteration 14601 / 49000) loss: 2.302558
(Epoch 60 / 200) train acc: 0.112000; val_acc: 0.102000
(Iteration 14701 / 49000) loss: 2.302769
(Iteration 14801 / 49000) loss: 2.302601
(Iteration 14901 / 49000) loss: 2.302712
(Epoch 61 / 200) train acc: 0.087000; val_acc: 0.102000
(Iteration 15001 / 49000) loss: 2.302124
(Iteration 15101 / 49000) loss: 2.302215
(Epoch 62 / 200) train acc: 0.092000; val_acc: 0.102000
(Iteration 15201 / 49000) loss: 2.302556
(Iteration 15301 / 49000) loss: 2.302919
(Iteration 15401 / 49000) loss: 2.303694
(Epoch 63 / 200) train acc: 0.098000; val_acc: 0.102000
(Iteration 15501 / 49000) loss: 2.302289
(Iteration 15601 / 49000) loss: 2.302886
(Epoch 64 / 200) train acc: 0.097000; val_acc: 0.102000
(Iteration 15701 / 49000) loss: 2.302314
(Iteration 15801 / 49000) loss: 2.302188
(Iteration 15901 / 49000) loss: 2.302750
(Epoch 65 / 200) train acc: 0.096000; val_acc: 0.087000
(Iteration 16001 / 49000) loss: 2.302491
(Iteration 16101 / 49000) loss: 2.302049
(Epoch 66 / 200) train acc: 0.104000; val_acc: 0.087000
```

```
(Iteration 16201 / 49000) loss: 2.302914
(Iteration 16301 / 49000) loss: 2.302295
(Iteration 16401 / 49000) loss: 2.302316
(Epoch 67 / 200) train acc: 0.106000; val_acc: 0.087000
(Iteration 16501 / 49000) loss: 2.302643
(Iteration 16601 / 49000) loss: 2.302762
(Epoch 68 / 200) train acc: 0.095000; val_acc: 0.113000
(Iteration 16701 / 49000) loss: 2.302782
(Iteration 16801 / 49000) loss: 2.302350
(Iteration 16901 / 49000) loss: 2.302221
(Epoch 69 / 200) train acc: 0.093000; val_acc: 0.112000
(Iteration 17001 / 49000) loss: 2.302105
(Iteration 17101 / 49000) loss: 2.302649
(Epoch 70 / 200) train acc: 0.086000; val_acc: 0.112000
(Iteration 17201 / 49000) loss: 2.302325
(Iteration 17301 / 49000) loss: 2.302581
(Epoch 71 / 200) train acc: 0.098000; val_acc: 0.112000
(Iteration 17401 / 49000) loss: 2.302796
(Iteration 17501 / 49000) loss: 2.302407
(Iteration 17601 / 49000) loss: 2.302901
(Epoch 72 / 200) train acc: 0.106000; val_acc: 0.112000
(Iteration 17701 / 49000) loss: 2.302528
(Iteration 17801 / 49000) loss: 2.303054
(Epoch 73 / 200) train acc: 0.105000; val_acc: 0.087000
(Iteration 17901 / 49000) loss: 2.302350
(Iteration 18001 / 49000) loss: 2.302121
(Iteration 18101 / 49000) loss: 2.302544
(Epoch 74 / 200) train acc: 0.099000; val_acc: 0.087000
(Iteration 18201 / 49000) loss: 2.302624
(Iteration 18301 / 49000) loss: 2.302981
(Epoch 75 / 200) train acc: 0.109000; val_acc: 0.087000
(Iteration 18401 / 49000) loss: 2.302330
(Iteration 18501 / 49000) loss: 2.301793
(Iteration 18601 / 49000) loss: 2.302624
(Epoch 76 / 200) train acc: 0.090000; val_acc: 0.087000
(Iteration 18701 / 49000) loss: 2.302647
(Iteration 18801 / 49000) loss: 2.302972
(Epoch 77 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 18901 / 49000) loss: 2.302481
(Iteration 19001 / 49000) loss: 2.302867
(Iteration 19101 / 49000) loss: 2.302749
(Epoch 78 / 200) train acc: 0.095000; val_acc: 0.087000
(Iteration 19201 / 49000) loss: 2.302074
(Iteration 19301 / 49000) loss: 2.302694
(Epoch 79 / 200) train acc: 0.090000; val_acc: 0.087000
(Iteration 19401 / 49000) loss: 2.302831
(Iteration 19501 / 49000) loss: 2.302472
(Epoch 80 / 200) train acc: 0.105000; val_acc: 0.079000
```

```
(Iteration 19601 / 49000) loss: 2.302690
(Iteration 19701 / 49000) loss: 2.302370
(Iteration 19801 / 49000) loss: 2.302793
(Epoch 81 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 19901 / 49000) loss: 2.302601
(Iteration 20001 / 49000) loss: 2.303021
(Epoch 82 / 200) train acc: 0.117000; val_acc: 0.112000
(Iteration 20101 / 49000) loss: 2.302527
(Iteration 20201 / 49000) loss: 2.302820
(Iteration 20301 / 49000) loss: 2.302572
(Epoch 83 / 200) train acc: 0.098000; val_acc: 0.112000
(Iteration 20401 / 49000) loss: 2.302771
(Iteration 20501 / 49000) loss: 2.302599
(Epoch 84 / 200) train acc: 0.096000; val_acc: 0.112000
(Iteration 20601 / 49000) loss: 2.302847
(Iteration 20701 / 49000) loss: 2.302792
(Iteration 20801 / 49000) loss: 2.302299
(Epoch 85 / 200) train acc: 0.110000; val_acc: 0.105000
(Iteration 20901 / 49000) loss: 2.302699
(Iteration 21001 / 49000) loss: 2.302580
(Epoch 86 / 200) train acc: 0.089000; val_acc: 0.105000
(Iteration 21101 / 49000) loss: 2.302400
(Iteration 21201 / 49000) loss: 2.302595
(Iteration 21301 / 49000) loss: 2.302331
(Epoch 87 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 21401 / 49000) loss: 2.302467
(Iteration 21501 / 49000) loss: 2.302560
(Epoch 88 / 200) train acc: 0.113000; val_acc: 0.087000
(Iteration 21601 / 49000) loss: 2.302530
(Iteration 21701 / 49000) loss: 2.302684
(Iteration 21801 / 49000) loss: 2.302444
(Epoch 89 / 200) train acc: 0.108000; val_acc: 0.087000
(Iteration 21901 / 49000) loss: 2.302339
(Iteration 22001 / 49000) loss: 2.302600
(Epoch 90 / 200) train acc: 0.098000; val_acc: 0.087000
(Iteration 22101 / 49000) loss: 2.302601
(Iteration 22201 / 49000) loss: 2.302634
(Epoch 91 / 200) train acc: 0.092000; val_acc: 0.087000
(Iteration 22301 / 49000) loss: 2.302642
(Iteration 22401 / 49000) loss: 2.302414
(Iteration 22501 / 49000) loss: 2.302553
(Epoch 92 / 200) train acc: 0.107000; val_acc: 0.112000
(Iteration 22601 / 49000) loss: 2.302302
(Iteration 22701 / 49000) loss: 2.302541
(Epoch 93 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 22801 / 49000) loss: 2.302914
(Iteration 22901 / 49000) loss: 2.302323
(Iteration 23001 / 49000) loss: 2.302076
```

```
(Epoch 94 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 23101 / 49000) loss: 2.302497
(Iteration 23201 / 49000) loss: 2.303030
(Epoch 95 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 23301 / 49000) loss: 2.302683
(Iteration 23401 / 49000) loss: 2.302864
(Iteration 23501 / 49000) loss: 2.302990
(Epoch 96 / 200) train acc: 0.107000; val_acc: 0.079000
(Iteration 23601 / 49000) loss: 2.302526
(Iteration 23701 / 49000) loss: 2.303046
(Epoch 97 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 23801 / 49000) loss: 2.302712
(Iteration 23901 / 49000) loss: 2.303023
(Iteration 24001 / 49000) loss: 2.302486
(Epoch 98 / 200) train acc: 0.115000; val_acc: 0.079000
(Iteration 24101 / 49000) loss: 2.302245
(Iteration 24201 / 49000) loss: 2.302513
(Epoch 99 / 200) train acc: 0.115000; val_acc: 0.107000
(Iteration 24301 / 49000) loss: 2.302434
(Iteration 24401 / 49000) loss: 2.302673
(Epoch 100 / 200) train acc: 0.118000; val_acc: 0.079000
(Iteration 24501 / 49000) loss: 2.302567
(Iteration 24601 / 49000) loss: 2.302806
(Iteration 24701 / 49000) loss: 2.302569
(Epoch 101 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 24801 / 49000) loss: 2.301931
(Iteration 24901 / 49000) loss: 2.302920
(Epoch 102 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 25001 / 49000) loss: 2.302293
(Iteration 25101 / 49000) loss: 2.302090
(Iteration 25201 / 49000) loss: 2.302557
(Epoch 103 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 25301 / 49000) loss: 2.302499
(Iteration 25401 / 49000) loss: 2.302641
(Epoch 104 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 25501 / 49000) loss: 2.302670
(Iteration 25601 / 49000) loss: 2.302712
(Iteration 25701 / 49000) loss: 2.302483
(Epoch 105 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 25801 / 49000) loss: 2.302455
(Iteration 25901 / 49000) loss: 2.302642
(Epoch 106 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 26001 / 49000) loss: 2.302448
(Iteration 26101 / 49000) loss: 2.302312
(Iteration 26201 / 49000) loss: 2.302803
(Epoch 107 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 26301 / 49000) loss: 2.302513
(Iteration 26401 / 49000) loss: 2.302592
```

```
(Epoch 108 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 26501 / 49000) loss: 2.302803
(Iteration 26601 / 49000) loss: 2.302245
(Iteration 26701 / 49000) loss: 2.302317
(Epoch 109 / 200) train acc: 0.088000; val_acc: 0.079000
(Iteration 26801 / 49000) loss: 2.302549
(Iteration 26901 / 49000) loss: 2.302580
(Epoch 110 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 27001 / 49000) loss: 2.302681
(Iteration 27101 / 49000) loss: 2.302667
(Epoch 111 / 200) train acc: 0.115000; val_acc: 0.079000
(Iteration 27201 / 49000) loss: 2.302996
(Iteration 27301 / 49000) loss: 2.302514
(Iteration 27401 / 49000) loss: 2.302699
(Epoch 112 / 200) train acc: 0.096000; val_acc: 0.079000
(Iteration 27501 / 49000) loss: 2.302498
(Iteration 27601 / 49000) loss: 2.302231
(Epoch 113 / 200) train acc: 0.088000; val_acc: 0.079000
(Iteration 27701 / 49000) loss: 2.302449
(Iteration 27801 / 49000) loss: 2.302489
(Iteration 27901 / 49000) loss: 2.302144
(Epoch 114 / 200) train acc: 0.104000; val_acc: 0.087000
(Iteration 28001 / 49000) loss: 2.302519
(Iteration 28101 / 49000) loss: 2.302966
(Epoch 115 / 200) train acc: 0.121000; val_acc: 0.079000
(Iteration 28201 / 49000) loss: 2.303093
(Iteration 28301 / 49000) loss: 2.302506
(Iteration 28401 / 49000) loss: 2.302608
(Epoch 116 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 28501 / 49000) loss: 2.302332
(Iteration 28601 / 49000) loss: 2.302482
(Epoch 117 / 200) train acc: 0.095000; val_acc: 0.079000
(Iteration 28701 / 49000) loss: 2.302635
(Iteration 28801 / 49000) loss: 2.302857
(Iteration 28901 / 49000) loss: 2.302302
(Epoch 118 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 29001 / 49000) loss: 2.302447
(Iteration 29101 / 49000) loss: 2.302227
(Epoch 119 / 200) train acc: 0.090000; val_acc: 0.079000
(Iteration 29201 / 49000) loss: 2.302741
(Iteration 29301 / 49000) loss: 2.303073
(Epoch 120 / 200) train acc: 0.099000; val_acc: 0.079000
(Iteration 29401 / 49000) loss: 2.302647
(Iteration 29501 / 49000) loss: 2.302251
(Iteration 29601 / 49000) loss: 2.302422
(Epoch 121 / 200) train acc: 0.102000; val_acc: 0.079000
(Iteration 29701 / 49000) loss: 2.302742
(Iteration 29801 / 49000) loss: 2.302025
```

```
(Epoch 122 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 29901 / 49000) loss: 2.302664
(Iteration 30001 / 49000) loss: 2.302339
(Iteration 30101 / 49000) loss: 2.302467
(Epoch 123 / 200) train acc: 0.079000; val_acc: 0.079000
(Iteration 30201 / 49000) loss: 2.302349
(Iteration 30301 / 49000) loss: 2.302821
(Epoch 124 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 30401 / 49000) loss: 2.302202
(Iteration 30501 / 49000) loss: 2.302923
(Iteration 30601 / 49000) loss: 2.302640
(Epoch 125 / 200) train acc: 0.116000; val_acc: 0.079000
(Iteration 30701 / 49000) loss: 2.302822
(Iteration 30801 / 49000) loss: 2.302773
(Epoch 126 / 200) train acc: 0.106000; val_acc: 0.079000
(Iteration 30901 / 49000) loss: 2.302431
(Iteration 31001 / 49000) loss: 2.302408
(Iteration 31101 / 49000) loss: 2.302318
(Epoch 127 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 31201 / 49000) loss: 2.302664
(Iteration 31301 / 49000) loss: 2.302374
(Epoch 128 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 31401 / 49000) loss: 2.302598
(Iteration 31501 / 49000) loss: 2.302483
(Iteration 31601 / 49000) loss: 2.302638
(Epoch 129 / 200) train acc: 0.109000; val_acc: 0.079000
(Iteration 31701 / 49000) loss: 2.302556
(Iteration 31801 / 49000) loss: 2.302580
(Epoch 130 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 31901 / 49000) loss: 2.302643
(Iteration 32001 / 49000) loss: 2.302703
(Epoch 131 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 32101 / 49000) loss: 2.302810
(Iteration 32201 / 49000) loss: 2.302489
(Iteration 32301 / 49000) loss: 2.302791
(Epoch 132 / 200) train acc: 0.088000; val_acc: 0.079000
(Iteration 32401 / 49000) loss: 2.302657
(Iteration 32501 / 49000) loss: 2.302365
(Epoch 133 / 200) train acc: 0.121000; val_acc: 0.079000
(Iteration 32601 / 49000) loss: 2.302281
(Iteration 32701 / 49000) loss: 2.302524
(Iteration 32801 / 49000) loss: 2.302716
(Epoch 134 / 200) train acc: 0.091000; val_acc: 0.079000
(Iteration 32901 / 49000) loss: 2.302408
(Iteration 33001 / 49000) loss: 2.302647
(Epoch 135 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 33101 / 49000) loss: 2.302676
(Iteration 33201 / 49000) loss: 2.302567
```

```
(Iteration 33301 / 49000) loss: 2.302668
(Epoch 136 / 200) train acc: 0.122000; val_acc: 0.079000
(Iteration 33401 / 49000) loss: 2.302575
(Iteration 33501 / 49000) loss: 2.302501
(Epoch 137 / 200) train acc: 0.101000; val_acc: 0.079000
(Iteration 33601 / 49000) loss: 2.302435
(Iteration 33701 / 49000) loss: 2.302614
(Iteration 33801 / 49000) loss: 2.302696
(Epoch 138 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 33901 / 49000) loss: 2.302530
(Iteration 34001 / 49000) loss: 2.302964
(Epoch 139 / 200) train acc: 0.098000; val_acc: 0.079000
(Iteration 34101 / 49000) loss: 2.302948
(Iteration 34201 / 49000) loss: 2.302811
(Epoch 140 / 200) train acc: 0.086000; val_acc: 0.079000
(Iteration 34301 / 49000) loss: 2.302501
(Iteration 34401 / 49000) loss: 2.302461
(Iteration 34501 / 49000) loss: 2.302406
(Epoch 141 / 200) train acc: 0.093000; val_acc: 0.079000
(Iteration 34601 / 49000) loss: 2.302683
(Iteration 34701 / 49000) loss: 2.302614
(Epoch 142 / 200) train acc: 0.084000; val_acc: 0.079000
(Iteration 34801 / 49000) loss: 2.303044
(Iteration 34901 / 49000) loss: 2.302689
(Iteration 35001 / 49000) loss: 2.302320
(Epoch 143 / 200) train acc: 0.104000; val_acc: 0.079000
(Iteration 35101 / 49000) loss: 2.302845
(Iteration 35201 / 49000) loss: 2.302820
(Epoch 144 / 200) train acc: 0.094000; val_acc: 0.079000
(Iteration 35301 / 49000) loss: 2.302628
(Iteration 35401 / 49000) loss: 2.302530
(Iteration 35501 / 49000) loss: 2.302323
(Epoch 145 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 35601 / 49000) loss: 2.302716
(Iteration 35701 / 49000) loss: 2.302577
(Epoch 146 / 200) train acc: 0.075000; val_acc: 0.079000
(Iteration 35801 / 49000) loss: 2.302328
(Iteration 35901 / 49000) loss: 2.302642
(Iteration 36001 / 49000) loss: 2.302308
(Epoch 147 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 36101 / 49000) loss: 2.302343
(Iteration 36201 / 49000) loss: 2.302535
(Epoch 148 / 200) train acc: 0.111000; val_acc: 0.079000
(Iteration 36301 / 49000) loss: 2.302560
(Iteration 36401 / 49000) loss: 2.302545
(Iteration 36501 / 49000) loss: 2.302382
(Epoch 149 / 200) train acc: 0.108000; val_acc: 0.079000
(Iteration 36601 / 49000) loss: 2.302608
```

```
(Iteration 36701 / 49000) loss: 2.302420
(Epoch 150 / 200) train acc: 0.105000; val_acc: 0.079000
(Iteration 36801 / 49000) loss: 2.302863
(Iteration 36901 / 49000) loss: 2.303117
(Epoch 151 / 200) train acc: 0.120000; val_acc: 0.079000
(Iteration 37001 / 49000) loss: 2.302634
(Iteration 37101 / 49000) loss: 2.302353
(Iteration 37201 / 49000) loss: 2.302582
(Epoch 152 / 200) train acc: 0.103000; val_acc: 0.079000
(Iteration 37301 / 49000) loss: 2.302021
(Iteration 37401 / 49000) loss: 2.302881
(Epoch 153 / 200) train acc: 0.100000; val_acc: 0.079000
(Iteration 37501 / 49000) loss: 2.302621
(Iteration 37601 / 49000) loss: 2.302635
(Iteration 37701 / 49000) loss: 2.302450
(Epoch 154 / 200) train acc: 0.101000; val_acc: 0.078000
(Iteration 37801 / 49000) loss: 2.302190
(Iteration 37901 / 49000) loss: 2.302349
(Epoch 155 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 38001 / 49000) loss: 2.302797
(Iteration 38101 / 49000) loss: 2.302666
(Iteration 38201 / 49000) loss: 2.302521
(Epoch 156 / 200) train acc: 0.089000; val_acc: 0.078000
(Iteration 38301 / 49000) loss: 2.302304
(Iteration 38401 / 49000) loss: 2.302728
(Epoch 157 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 38501 / 49000) loss: 2.301928
(Iteration 38601 / 49000) loss: 2.302397
(Iteration 38701 / 49000) loss: 2.302591
(Epoch 158 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 38801 / 49000) loss: 2.302549
(Iteration 38901 / 49000) loss: 2.302633
(Epoch 159 / 200) train acc: 0.091000; val_acc: 0.078000
(Iteration 39001 / 49000) loss: 2.302427
(Iteration 39101 / 49000) loss: 2.302179
(Epoch 160 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 39201 / 49000) loss: 2.302812
(Iteration 39301 / 49000) loss: 2.302793
(Iteration 39401 / 49000) loss: 2.302445
(Epoch 161 / 200) train acc: 0.117000; val_acc: 0.078000
(Iteration 39501 / 49000) loss: 2.302450
(Iteration 39601 / 49000) loss: 2.302400
(Epoch 162 / 200) train acc: 0.105000; val_acc: 0.078000
(Iteration 39701 / 49000) loss: 2.302639
(Iteration 39801 / 49000) loss: 2.302479
(Iteration 39901 / 49000) loss: 2.302879
(Epoch 163 / 200) train acc: 0.087000; val_acc: 0.078000
(Iteration 40001 / 49000) loss: 2.302039
```

```
(Iteration 40101 / 49000) loss: 2.302991
(Epoch 164 / 200) train acc: 0.090000; val_acc: 0.078000
(Iteration 40201 / 49000) loss: 2.302512
(Iteration 40301 / 49000) loss: 2.302674
(Iteration 40401 / 49000) loss: 2.302604
(Epoch 165 / 200) train acc: 0.131000; val_acc: 0.078000
(Iteration 40501 / 49000) loss: 2.302380
(Iteration 40601 / 49000) loss: 2.302696
(Epoch 166 / 200) train acc: 0.109000; val_acc: 0.078000
(Iteration 40701 / 49000) loss: 2.302722
(Iteration 40801 / 49000) loss: 2.302407
(Iteration 40901 / 49000) loss: 2.302589
(Epoch 167 / 200) train acc: 0.096000; val_acc: 0.078000
(Iteration 41001 / 49000) loss: 2.302176
(Iteration 41101 / 49000) loss: 2.302817
(Epoch 168 / 200) train acc: 0.092000; val_acc: 0.078000
(Iteration 41201 / 49000) loss: 2.302312
(Iteration 41301 / 49000) loss: 2.302690
(Iteration 41401 / 49000) loss: 2.302662
(Epoch 169 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 41501 / 49000) loss: 2.302021
(Iteration 41601 / 49000) loss: 2.302656
(Epoch 170 / 200) train acc: 0.100000; val_acc: 0.078000
(Iteration 41701 / 49000) loss: 2.302336
(Iteration 41801 / 49000) loss: 2.302043
(Epoch 171 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 41901 / 49000) loss: 2.302204
(Iteration 42001 / 49000) loss: 2.303013
(Iteration 42101 / 49000) loss: 2.302106
(Epoch 172 / 200) train acc: 0.093000; val_acc: 0.078000
(Iteration 42201 / 49000) loss: 2.302711
(Iteration 42301 / 49000) loss: 2.302430
(Epoch 173 / 200) train acc: 0.099000; val_acc: 0.078000
(Iteration 42401 / 49000) loss: 2.303110
(Iteration 42501 / 49000) loss: 2.303375
(Iteration 42601 / 49000) loss: 2.302995
(Epoch 174 / 200) train acc: 0.097000; val_acc: 0.078000
(Iteration 42701 / 49000) loss: 2.302803
(Iteration 42801 / 49000) loss: 2.302870
(Epoch 175 / 200) train acc: 0.085000; val_acc: 0.078000
(Iteration 42901 / 49000) loss: 2.302575
(Iteration 43001 / 49000) loss: 2.302497
(Iteration 43101 / 49000) loss: 2.303031
(Epoch 176 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 43201 / 49000) loss: 2.302549
(Iteration 43301 / 49000) loss: 2.302757
(Epoch 177 / 200) train acc: 0.114000; val_acc: 0.078000
(Iteration 43401 / 49000) loss: 2.302856
```

```
(Iteration 43501 / 49000) loss: 2.302832
(Iteration 43601 / 49000) loss: 2.302946
(Epoch 178 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 43701 / 49000) loss: 2.302540
(Iteration 43801 / 49000) loss: 2.302441
(Epoch 179 / 200) train acc: 0.111000; val_acc: 0.078000
(Iteration 43901 / 49000) loss: 2.302488
(Iteration 44001 / 49000) loss: 2.303119
(Epoch 180 / 200) train acc: 0.111000; val_acc: 0.078000
(Iteration 44101 / 49000) loss: 2.302311
(Iteration 44201 / 49000) loss: 2.302620
(Iteration 44301 / 49000) loss: 2.302786
(Epoch 181 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 44401 / 49000) loss: 2.302647
(Iteration 44501 / 49000) loss: 2.302358
(Epoch 182 / 200) train acc: 0.102000; val_acc: 0.078000
(Iteration 44601 / 49000) loss: 2.302286
(Iteration 44701 / 49000) loss: 2.302652
(Iteration 44801 / 49000) loss: 2.303036
(Epoch 183 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 44901 / 49000) loss: 2.302748
(Iteration 45001 / 49000) loss: 2.302572
(Epoch 184 / 200) train acc: 0.114000; val_acc: 0.078000
(Iteration 45101 / 49000) loss: 2.302314
(Iteration 45201 / 49000) loss: 2.302368
(Iteration 45301 / 49000) loss: 2.301906
(Epoch 185 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 45401 / 49000) loss: 2.302826
(Iteration 45501 / 49000) loss: 2.302554
(Epoch 186 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 45601 / 49000) loss: 2.302162
(Iteration 45701 / 49000) loss: 2.302525
(Iteration 45801 / 49000) loss: 2.301944
(Epoch 187 / 200) train acc: 0.098000; val_acc: 0.078000
(Iteration 45901 / 49000) loss: 2.302832
(Iteration 46001 / 49000) loss: 2.302348
(Epoch 188 / 200) train acc: 0.094000; val_acc: 0.078000
(Iteration 46101 / 49000) loss: 2.302368
(Iteration 46201 / 49000) loss: 2.302378
(Iteration 46301 / 49000) loss: 2.302653
(Epoch 189 / 200) train acc: 0.112000; val_acc: 0.078000
(Iteration 46401 / 49000) loss: 2.303272
(Iteration 46501 / 49000) loss: 2.302539
(Epoch 190 / 200) train acc: 0.089000; val_acc: 0.078000
(Iteration 46601 / 49000) loss: 2.302527
(Iteration 46701 / 49000) loss: 2.302574
(Epoch 191 / 200) train acc: 0.083000; val_acc: 0.078000
(Iteration 46801 / 49000) loss: 2.302330
```

```
(Iteration 46901 / 49000) loss: 2.302572
(Iteration 47001 / 49000) loss: 2.302735
(Epoch 192 / 200) train acc: 0.104000; val_acc: 0.078000
(Iteration 47101 / 49000) loss: 2.302541
(Iteration 47201 / 49000) loss: 2.302816
(Epoch 193 / 200) train acc: 0.111000; val_acc: 0.078000
(Iteration 47301 / 49000) loss: 2.302545
(Iteration 47401 / 49000) loss: 2.302630
(Iteration 47501 / 49000) loss: 2.302768
(Epoch 194 / 200) train acc: 0.095000; val_acc: 0.078000
(Iteration 47601 / 49000) loss: 2.302749
(Iteration 47701 / 49000) loss: 2.302091
(Epoch 195 / 200) train acc: 0.110000; val_acc: 0.078000
(Iteration 47801 / 49000) loss: 2.302590
(Iteration 47901 / 49000) loss: 2.302577
(Iteration 48001 / 49000) loss: 2.302821
(Epoch 196 / 200) train acc: 0.107000; val_acc: 0.078000
(Iteration 48101 / 49000) loss: 2.302514
(Iteration 48201 / 49000) loss: 2.302212
(Epoch 197 / 200) train acc: 0.086000; val_acc: 0.078000
(Iteration 48301 / 49000) loss: 2.302391
(Iteration 48401 / 49000) loss: 2.302576
(Iteration 48501 / 49000) loss: 2.302404
(Epoch 198 / 200) train acc: 0.106000; val_acc: 0.078000
(Iteration 48601 / 49000) loss: 2.302795
(Iteration 48701 / 49000) loss: 2.302465
(Epoch 199 / 200) train acc: 0.099000; val_acc: 0.078000
(Iteration 48801 / 49000) loss: 2.302560
(Iteration 48901 / 49000) loss: 2.302747
(Epoch 200 / 200) train acc: 0.134000; val_acc: 0.078000
```

[11]:
```python
# Run your best neural net classifier on the test set. You should be able
# to get more than 58% accuracy. It is also possible to get >60% accuracy
# with careful tuning.

y_test_pred = np.argmax(best_net.loss(data['X_test']), axis=1)
test_acc = (y_test_pred == data['y_test']).mean()
print(test_acc)
```

```
0.377
```

[12]:
```python
# Save best model
best_net.save("best_two_layer_net_features.npy")
```

```
best_two_layer_net_features.npy saved.
```

[12]:

FullyConnectedNets

December 18, 2025

```
[1]: # This mounts your Google Drive to the Colab VM.
     from google.colab import drive
     drive.mount('/content/drive')

     # TODO: Enter the foldername in your Drive where you have saved the unzipped
     # assignment folder, e.g. 'icv83551/assignments/assignment1/'
     FOLDERNAME = 'icv83551/assignments/assignment1/'
     assert FOLDERNAME is not None, "[!] Enter the foldername."

     # Now that we've mounted your Drive, this ensures that
     # the Python interpreter of the Colab VM can load
     # python files from within it.
     import sys
     sys.path.append('/content/drive/My Drive/{}'.format(FOLDERNAME))

     # This downloads the CIFAR-10 dataset to your Drive
     # if it doesn't already exist.
     %cd /content/drive/My\ Drive/$FOLDERNAME/icv83551/datasets/
     # %cd /content/drive/My\ Drive/$FOLDERNAME
     !bash get_datasets.sh
     %cd /content/drive/My\ Drive/$FOLDERNAME
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call
drive.mount("/content/drive", force_remount=True).
/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/datasets
/content/drive/My Drive/icv83551/assignments/assignment1
```

1 Multi-Layer Fully Connected Network

In this exercise, you will implement a fully connected network with an arbitrary number of hidden layers.

```
[2]: # from google.colab import drive
     # drive.mount('/content/drive')
```

Read through the `FullyConnectedNet` class in the file `icv83551/classifiers/fc_net.py`.

Implement the network initialization, forward pass, and backward pass. Throughout this assign-

1

ment, you will be implementing layers in `icv83551/layers.py`. You can re-use your implementations for `affine_forward`, `affine_backward`, `relu_forward`, `relu_backward`, and `softmax_loss` from before. For right now, don't worry about implementing dropout or batch/layer normalization yet, as you will add those features later.

```
[3]: # Setup cell.
     import time
     import numpy as np
     import matplotlib.pyplot as plt
     from icv83551.classifiers.fc_net import *
     from icv83551.data_utils import get_CIFAR10_data
     from icv83551.gradient_check import eval_numerical_gradient,␣
       ↪eval_numerical_gradient_array
     from icv83551.solver import Solver

     %matplotlib inline
     plt.rcParams["figure.figsize"] = (10.0, 8.0)  # Set default size of plots.
     plt.rcParams["image.interpolation"] = "nearest"
     plt.rcParams["image.cmap"] = "gray"

     #%load_ext autoreload
     #%autoreload 2

     def rel_error(x, y):
         """Returns relative error."""
         return np.max(np.abs(x - y) / (np.maximum(1e-8, np.abs(x) + np.abs(y))))
```

```
[4]: # Load the (preprocessed) CIFAR-10 data.
     data = get_CIFAR10_data()
     for k, v in list(data.items()):
         print(f"{k}: {v.shape}")
```

```
X_train: (49000, 3, 32, 32)
y_train: (49000,)
X_val: (1000, 3, 32, 32)
y_val: (1000,)
X_test: (1000, 3, 32, 32)
y_test: (1000,)
```

1.1 Initial Loss and Gradient Check

As a sanity check, run the following to check the initial loss and to gradient check the network both with and without regularization. This is a good way to see if the initial losses seem reasonable.

For gradient checking, you should expect to see errors around 1e-7 or less.

```
[5]: np.random.seed(231)
     N, D, H1, H2, C = 2, 15, 20, 30, 10
     X = np.random.randn(N, D)
```

```python
y = np.random.randint(C, size=(N,))

for reg in [0, 3.14]:
    print("Running check with reg = ", reg)
    model = FullyConnectedNet(
        [H1, H2],
        input_dim=D,
        num_classes=C,
        reg=reg,
        weight_scale=5e-2,
        dtype=np.float64
    )

    loss, grads = model.loss(X, y)
    print("Initial loss: ", loss)

    # Most of the errors should be on the order of e-7 or smaller.
    # NOTE: It is fine however to see an error for W2 on the order of e-5
    # for the check when reg = 0.0
    for name in sorted(grads):
        f = lambda _: model.loss(X, y)[0]
        grad_num = eval_numerical_gradient(f, model.params[name],␣
  ↪verbose=False, h=1e-5)
        print(f"{name} relative error: {rel_error(grad_num, grads[name])}")
```

```
Running check with reg =  0
Initial loss:  2.300479089768492
W1 relative error: 1.0252674471656573e-07
W2 relative error: 2.2120479295080622e-05
W3 relative error: 4.5623278736665505e-07
b1 relative error: 4.6600944653202505e-09
b2 relative error: 2.085654276112763e-09
b3 relative error: 1.689724888469736e-10
Running check with reg =  3.14
Initial loss:  7.052114776533016
W1 relative error: 6.862884860440611e-09
W2 relative error: 3.522821562176466e-08
W3 relative error: 2.6171457283983532e-08
b1 relative error: 1.4752427965311745e-08
b2 relative error: 1.7223751746766738e-09
b3 relative error: 2.378772438198909e-10
```

As another sanity check, make sure your network can overfit on a small dataset of 50 images. First, we will try a three-layer network with 100 units in each hidden layer. In the following cell, tweak the **learning rate** and **weight initialization scale** to overfit and achieve 100% training accuracy within 20 epochs.

```
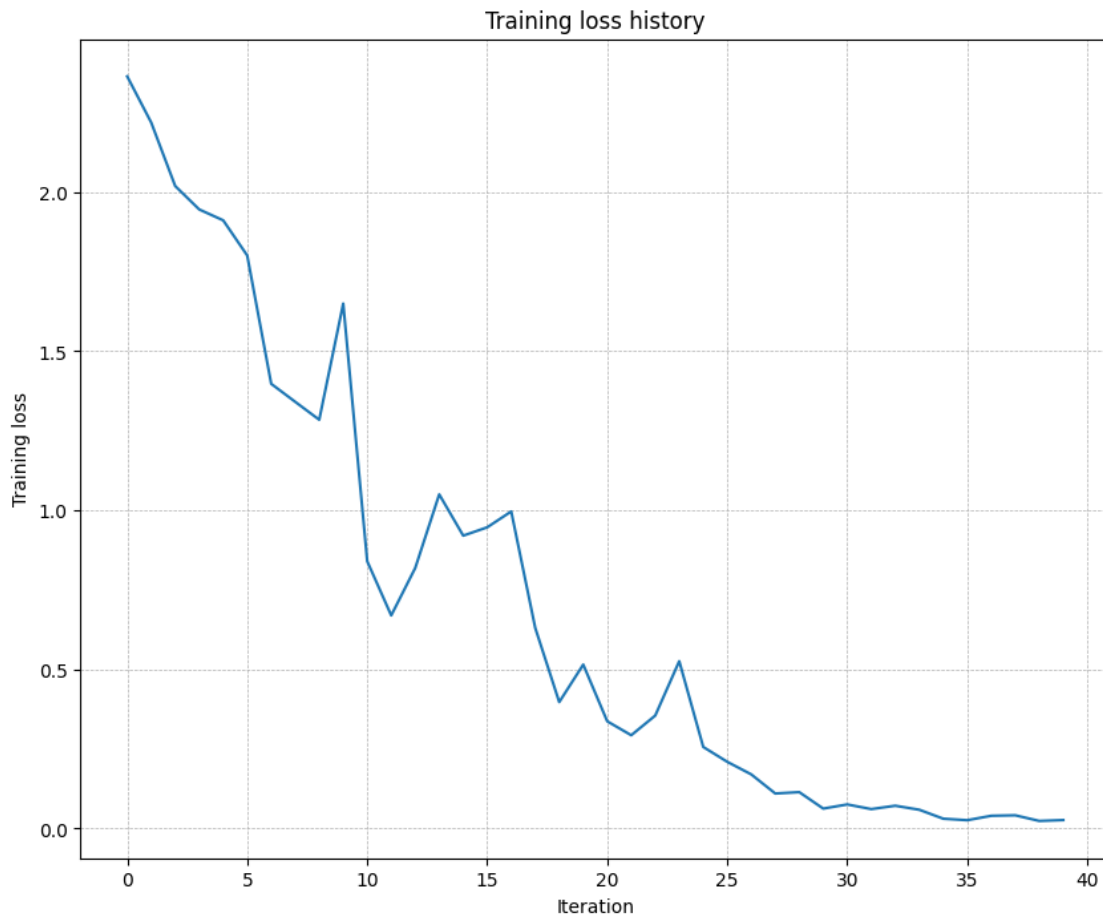[6]:  # TODO: Use a three-layer Net to overfit 50 training examples by
      # tweaking just the learning rate and initialization scale.

      num_train = 50
      small_data = {
          "X_train": data["X_train"][:num_train],
          "y_train": data["y_train"][:num_train],
          "X_val": data["X_val"],
          "y_val": data["y_val"],
      }

      weight_scale = 1e-2
      learning_rate = 1e-2


      model = FullyConnectedNet(
          [100, 100],
          weight_scale=weight_scale,
          dtype=np.float64
      )
      solver = Solver(
          model,
          small_data,
          print_every=10,
          num_epochs=20,
          batch_size=25,
          update_rule="sgd",
          optim_config={"learning_rate": learning_rate},
      )
      solver.train()

      plt.plot(solver.loss_history)
      plt.title("Training loss history")
      plt.xlabel("Iteration")
      plt.ylabel("Training loss")
      plt.grid(linestyle='--', linewidth=0.5)
      plt.show()
```

```
(Iteration 1 / 40) loss: 2.363364
(Epoch 0 / 20) train acc: 0.180000; val_acc: 0.108000
(Epoch 1 / 20) train acc: 0.320000; val_acc: 0.127000
(Epoch 2 / 20) train acc: 0.440000; val_acc: 0.172000
(Epoch 3 / 20) train acc: 0.500000; val_acc: 0.184000
(Epoch 4 / 20) train acc: 0.540000; val_acc: 0.181000
(Epoch 5 / 20) train acc: 0.740000; val_acc: 0.190000
(Iteration 11 / 40) loss: 0.839976
(Epoch 6 / 20) train acc: 0.740000; val_acc: 0.187000
```

```
(Epoch 7 / 20) train acc: 0.740000; val_acc: 0.183000
(Epoch 8 / 20) train acc: 0.820000; val_acc: 0.177000
(Epoch 9 / 20) train acc: 0.860000; val_acc: 0.200000
(Epoch 10 / 20) train acc: 0.920000; val_acc: 0.191000
(Iteration 21 / 40) loss: 0.337174
(Epoch 11 / 20) train acc: 0.960000; val_acc: 0.189000
(Epoch 12 / 20) train acc: 0.940000; val_acc: 0.180000
(Epoch 13 / 20) train acc: 1.000000; val_acc: 0.199000
(Epoch 14 / 20) train acc: 1.000000; val_acc: 0.199000
(Epoch 15 / 20) train acc: 1.000000; val_acc: 0.195000
(Iteration 31 / 40) loss: 0.075911
(Epoch 16 / 20) train acc: 1.000000; val_acc: 0.182000
(Epoch 17 / 20) train acc: 1.000000; val_acc: 0.201000
(Epoch 18 / 20) train acc: 1.000000; val_acc: 0.207000
(Epoch 19 / 20) train acc: 1.000000; val_acc: 0.185000
(Epoch 20 / 20) train acc: 1.000000; val_acc: 0.192000
```


Training loss history

Now, try to use a five-layer network with 100 units on each layer to overfit on 50 training examples. Again, you will have to adjust the learning rate and weight initialization scale, but you should be

able to achieve 100% training accuracy within 20 epochs.

```python
[7]: # TODO: Use a five-layer Net to overfit 50 training examples by
     # tweaking just the learning rate and initialization scale.

     num_train = 50
     small_data = {
        'X_train': data['X_train'][:num_train],
        'y_train': data['y_train'][:num_train],
        'X_val': data['X_val'],
        'y_val': data['y_val'],
     }

     learning_rate = 0.02 # Experiment with this!
     weight_scale = 0.05  # Experiment with this!


     model = FullyConnectedNet(
         [100, 100, 100, 100],
         weight_scale=weight_scale,
         dtype=np.float64
     )
     solver = Solver(
         model,
         small_data,
         print_every=10,
         num_epochs=20,
         batch_size=25,
         update_rule='sgd',
         optim_config={'learning_rate': learning_rate},
     )
     solver.train()

     plt.plot(solver.loss_history)
     plt.title('Training loss history')
     plt.xlabel('Iteration')
     plt.ylabel('Training loss')
     plt.grid(linestyle='--', linewidth=0.5)
     plt.show()
```

```
(Iteration 1 / 40) loss: 5.557032
(Epoch 0 / 20) train acc: 0.080000; val_acc: 0.096000
(Epoch 1 / 20) train acc: 0.320000; val_acc: 0.117000
(Epoch 2 / 20) train acc: 0.180000; val_acc: 0.104000
(Epoch 3 / 20) train acc: 0.560000; val_acc: 0.124000
(Epoch 4 / 20) train acc: 0.640000; val_acc: 0.103000
(Epoch 5 / 20) train acc: 0.680000; val_acc: 0.143000
(Iteration 11 / 40) loss: 1.127344
```

```
(Epoch 6 / 20) train acc: 0.740000; val_acc: 0.140000
(Epoch 7 / 20) train acc: 0.820000; val_acc: 0.152000
(Epoch 8 / 20) train acc: 0.940000; val_acc: 0.133000
(Epoch 9 / 20) train acc: 0.920000; val_acc: 0.138000
(Epoch 10 / 20) train acc: 0.940000; val_acc: 0.142000
(Iteration 21 / 40) loss: 0.273782
(Epoch 11 / 20) train acc: 0.960000; val_acc: 0.136000
(Epoch 12 / 20) train acc: 0.960000; val_acc: 0.143000
(Epoch 13 / 20) train acc: 0.980000; val_acc: 0.138000
(Epoch 14 / 20) train acc: 1.000000; val_acc: 0.137000
(Epoch 15 / 20) train acc: 1.000000; val_acc: 0.137000
(Iteration 31 / 40) loss: 0.085884
(Epoch 16 / 20) train acc: 1.000000; val_acc: 0.138000
(Epoch 17 / 20) train acc: 1.000000; val_acc: 0.137000
(Epoch 18 / 20) train acc: 1.000000; val_acc: 0.137000
(Epoch 19 / 20) train acc: 1.000000; val_acc: 0.135000
(Epoch 20 / 20) train acc: 1.000000; val_acc: 0.141000
```


Training loss history
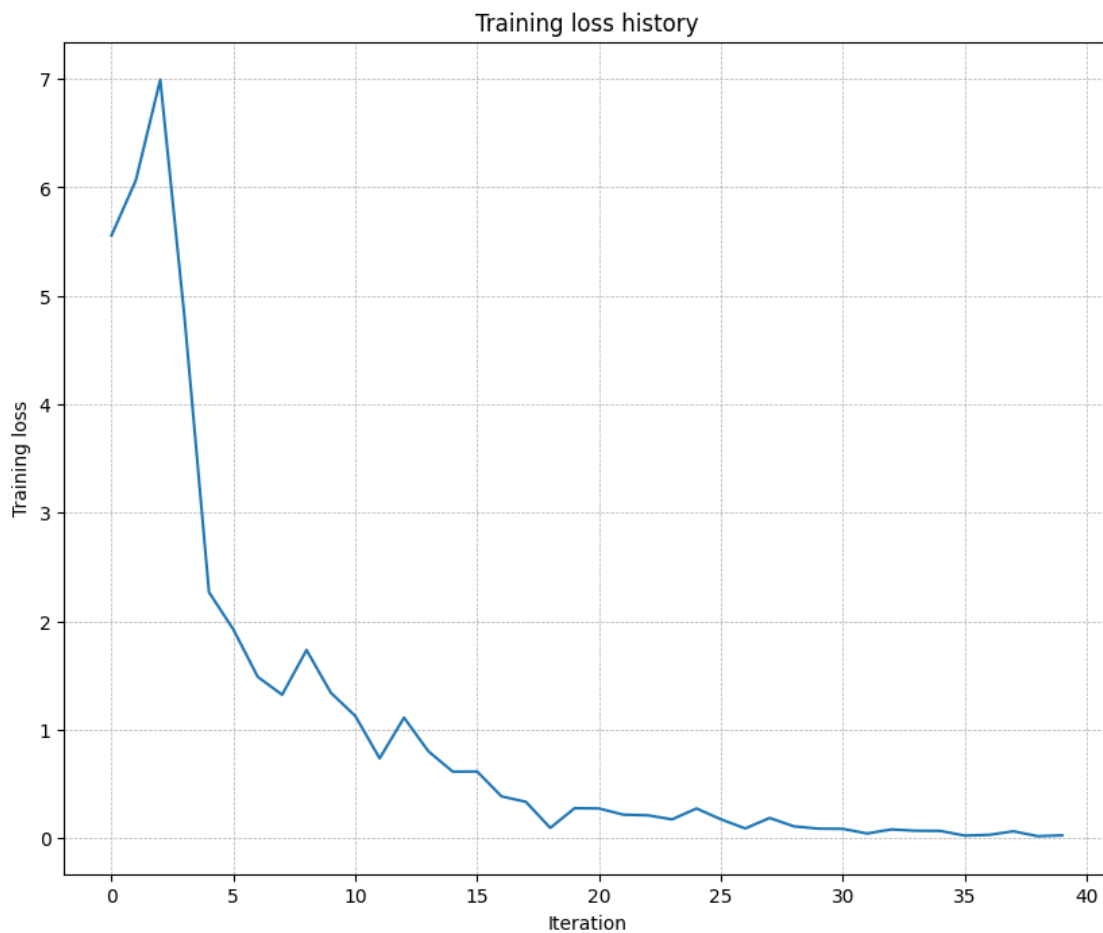
1.2 Inline Question 1:

Did you notice anything about the comparative difficulty of training the three-layer network vs. training the five-layer network? In particular, based on your experience, which network seemed more sensitive to the initialization scale? Why do you think that is the case?

1.3 Answer:

the 5 - layer is much more sensetive as we multiply the input data 5 times, meaning if its to big or small we will get the issue of vanishing/exploding activation, same as to the backward pass with vanishing/exploding graident. meaning, its harded to get right the 5 layer, and its more difficult to train.

2 Update rules

So far we have used vanilla stochastic gradient descent (SGD) as our update rule. More sophisticated update rules can make it easier to train deep networks. We will implement a few of the most commonly used update rules and compare them to vanilla SGD.

2.1 SGD+Momentum

Stochastic gradient descent with momentum is a widely used update rule that tends to make deep networks converge faster than vanilla stochastic gradient descent.

Open the file `icv83551/optim.py` and read the documentation at the top of the file to make sure you understand the API. Implement the SGD+momentum update rule in the function `sgd_momentum` and run the following to check your implementation. You should see errors less than e-8.

```python
[8]: from icv83551.optim import sgd_momentum

N, D = 4, 5
w = np.linspace(-0.4, 0.6, num=N*D).reshape(N, D)
dw = np.linspace(-0.6, 0.4, num=N*D).reshape(N, D)
v = np.linspace(0.6, 0.9, num=N*D).reshape(N, D)

config = {"learning_rate": 1e-3, "velocity": v}
next_w, _ = sgd_momentum(w, dw, config=config)

expected_next_w = np.asarray([
  [ 0.1406,      0.20738947,  0.27417895,  0.34096842,  0.40775789],
  [ 0.47454737,  0.54133684,  0.60812632,  0.67491579,  0.74170526],
  [ 0.80849474,  0.87528421,  0.94207368,  1.00886316,  1.07565263],
  [ 1.14244211,  1.20923158,  1.27602105,  1.34281053,  1.4096     ]])
expected_velocity = np.asarray([
  [ 0.5406,      0.55475789,  0.56891579,  0.58307368,  0.59723158],
  [ 0.61138947,  0.62554737,  0.63970526,  0.65386316,  0.66802105],
  [ 0.68217895,  0.69633684,  0.71049474,  0.72465263,  0.73881053],
  [ 0.75296842,  0.76712632,  0.78128421,  0.79544211,  0.8096     ]])
```

```
# Should see relative errors around e-8 or less
print("next_w error: ", rel_error(next_w, expected_next_w))
print("velocity error: ", rel_error(expected_velocity, config["velocity"]))
```

```
next_w error:  8.882347033505819e-09
velocity error:  4.269287743278663e-09
```

Once you have done so, run the following to train a six-layer network with both SGD and SGD+momentum. You should see the SGD+momentum update rule converge faster.

```
[9]: num_train = 4000
small_data = {
    'X_train': data['X_train'][:num_train],
    'y_train': data['y_train'][:num_train],
    'X_val': data['X_val'],
    'y_val': data['y_val'],
}

solvers = {}

for update_rule in ['sgd', 'sgd_momentum']:
    print('Running with ', update_rule)
    model = FullyConnectedNet(
        [100, 100, 100, 100, 100],
        weight_scale=5e-2
    )

    solver = Solver(
        model,
        small_data,
        num_epochs=5,
        batch_size=100,
        update_rule=update_rule,
        optim_config={'learning_rate': 5e-3},
        verbose=True,
    )
    solvers[update_rule] = solver
    solver.train()

fig, axes = plt.subplots(3, 1, figsize=(15, 15))

axes[0].set_title('Training loss')
axes[0].set_xlabel('Iteration')
axes[1].set_title('Training accuracy')
axes[1].set_xlabel('Epoch')
axes[2].set_title('Validation accuracy')
axes[2].set_xlabel('Epoch')
```

```python
for update_rule, solver in solvers.items():
    axes[0].plot(solver.loss_history, label=f"loss_{update_rule}")
    axes[1].plot(solver.train_acc_history, label=f"train_acc_{update_rule}")
    axes[2].plot(solver.val_acc_history, label=f"val_acc_{update_rule}")

for ax in axes:
    ax.legend(loc="best", ncol=4)
    ax.grid(linestyle='--', linewidth=0.5)

plt.show()
```

```
Running with  sgd
(Iteration 1 / 200) loss: 2.559978
(Epoch 0 / 5) train acc: 0.104000; val_acc: 0.107000
(Iteration 11 / 200) loss: 2.356070
(Iteration 21 / 200) loss: 2.214091
(Iteration 31 / 200) loss: 2.205928
(Epoch 1 / 5) train acc: 0.225000; val_acc: 0.193000
(Iteration 41 / 200) loss: 2.132095
(Iteration 51 / 200) loss: 2.118950
(Iteration 61 / 200) loss: 2.116443
(Iteration 71 / 200) loss: 2.132549
(Epoch 2 / 5) train acc: 0.298000; val_acc: 0.260000
(Iteration 81 / 200) loss: 1.977227
(Iteration 91 / 200) loss: 2.007528
(Iteration 101 / 200) loss: 2.004762
(Iteration 111 / 200) loss: 1.885342
(Epoch 3 / 5) train acc: 0.343000; val_acc: 0.287000
(Iteration 121 / 200) loss: 1.891516
(Iteration 131 / 200) loss: 1.923677
(Iteration 141 / 200) loss: 1.957743
(Iteration 151 / 200) loss: 1.966736
(Epoch 4 / 5) train acc: 0.322000; val_acc: 0.305000
(Iteration 161 / 200) loss: 1.801483
(Iteration 171 / 200) loss: 1.973780
(Iteration 181 / 200) loss: 1.666573
(Iteration 191 / 200) loss: 1.909494
(Epoch 5 / 5) train acc: 0.372000; val_acc: 0.319000
Running with  sgd_momentum
(Iteration 1 / 200) loss: 3.153778
(Epoch 0 / 5) train acc: 0.099000; val_acc: 0.088000
(Iteration 11 / 200) loss: 2.227203
(Iteration 21 / 200) loss: 2.125706
(Iteration 31 / 200) loss: 1.932695
(Epoch 1 / 5) train acc: 0.307000; val_acc: 0.260000
(Iteration 41 / 200) loss: 1.946488
(Iteration 51 / 200) loss: 1.778583
```

```
(Iteration 61 / 200) loss: 1.758119
(Iteration 71 / 200) loss: 1.849137
(Epoch 2 / 5) train acc: 0.382000; val_acc: 0.322000
(Iteration 81 / 200) loss: 2.048671
(Iteration 91 / 200) loss: 1.693223
(Iteration 101 / 200) loss: 1.511693
(Iteration 111 / 200) loss: 1.390754
(Epoch 3 / 5) train acc: 0.458000; val_acc: 0.338000
(Iteration 121 / 200) loss: 1.670614
(Iteration 131 / 200) loss: 1.540271
(Iteration 141 / 200) loss: 1.597365
(Iteration 151 / 200) loss: 1.609851
(Epoch 4 / 5) train acc: 0.490000; val_acc: 0.327000
(Iteration 161 / 200) loss: 1.472687
(Iteration 171 / 200) loss: 1.378620
(Iteration 181 / 200) loss: 1.378175
(Iteration 191 / 200) loss: 1.305934
(Epoch 5 / 5) train acc: 0.536000; val_acc: 0.368000
```
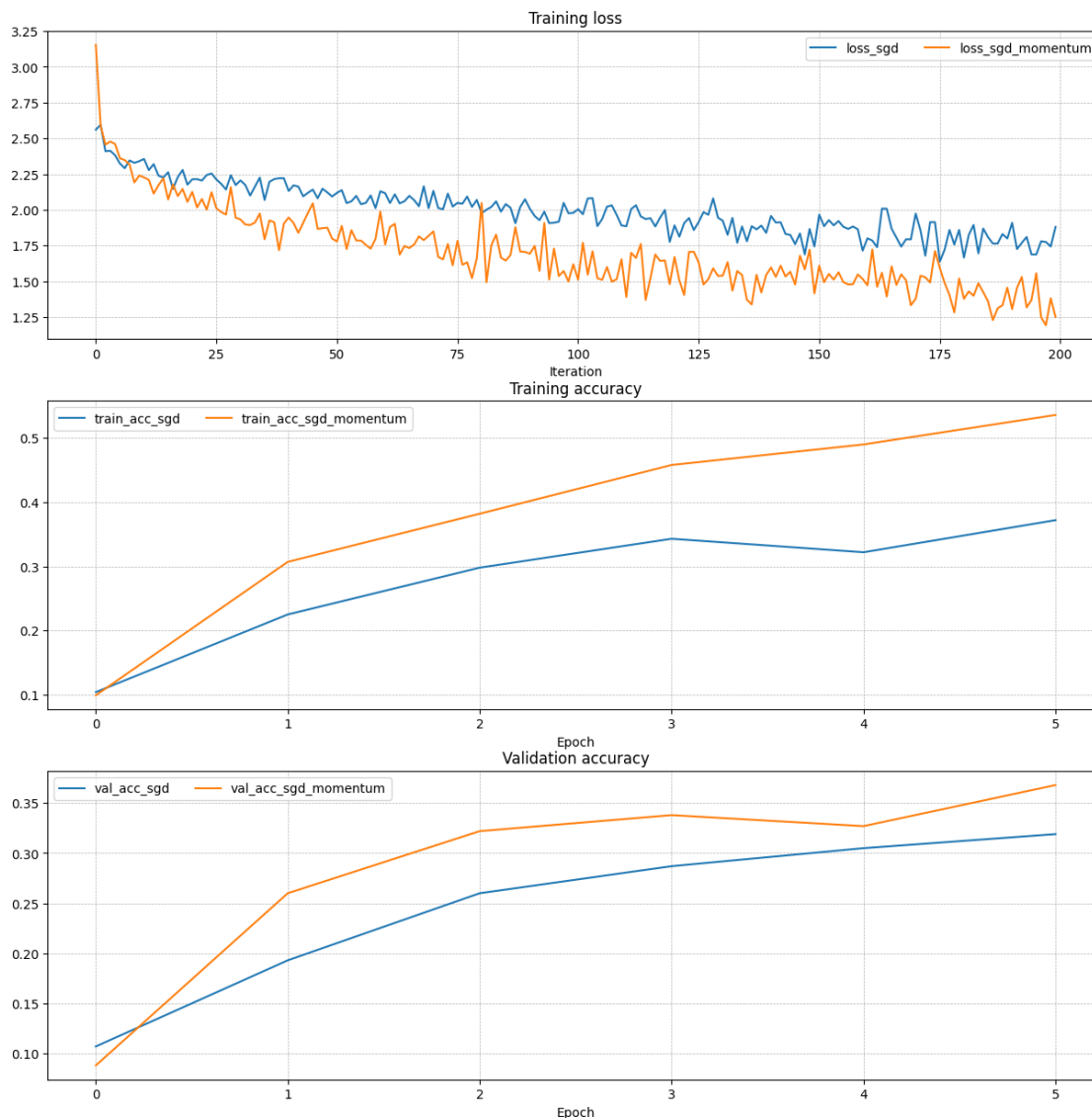
2.2 RMSProp and Adam

RMSProp [1] and Adam [2] are update rules that set per-parameter learning rates by using a running average of the second moments of gradients.

In the file `icv83551/optim.py`, implement the RMSProp update rule in the `rmsprop` function and implement the Adam update rule in the `adam` function, and check your implementations using the tests below.

NOTE: Please implement the *complete* Adam update rule (with the bias correction mechanism), not the first simplified version mentioned in the course notes.

[1] Tijmen Tieleman and Geoffrey Hinton. "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude." COURSERA: Neural Networks for Machine Learning 4 (2012).

[2] Diederik Kingma and Jimmy Ba, "Adam: A Method for Stochastic Optimization", ICLR 2015.

```python
[10]: # Test RMSProp implementation
      from icv83551.optim import rmsprop

      N, D = 4, 5
      w = np.linspace(-0.4, 0.6, num=N*D).reshape(N, D)
      dw = np.linspace(-0.6, 0.4, num=N*D).reshape(N, D)
      cache = np.linspace(0.6, 0.9, num=N*D).reshape(N, D)

      config = {'learning_rate': 1e-2, 'cache': cache}
      next_w, _ = rmsprop(w, dw, config=config)

      expected_next_w = np.asarray([
        [-0.39223849, -0.34037513, -0.28849239, -0.23659121, -0.18467247],
        [-0.132737,   -0.08078555, -0.02881884,  0.02316247,  0.07515774],
        [ 0.12716641,  0.17918792,  0.23122175,  0.28326742,  0.33532447],
        [ 0.38739248,  0.43947102,  0.49155973,  0.54365823,  0.59576619]])
      expected_cache = np.asarray([
        [ 0.5976,      0.6126277,   0.6277108,   0.64284931,  0.65804321],
        [ 0.67329252,  0.68859723,  0.70395734,  0.71937285,  0.73484377],
        [ 0.75037008,  0.7659518,   0.78158892,  0.79728144,  0.81302936],
        [ 0.82883269,  0.84469141,  0.86060554,  0.87657507,  0.8926    ]])

      # You should see relative errors around e-7 or less
      print('next_w error: ', rel_error(expected_next_w, next_w))
      print('cache error: ', rel_error(expected_cache, config['cache']))
```

```
next_w error:  9.524687511038133e-08
cache error:  2.6477955807156126e-09
```

```python
[11]: # Test Adam implementation
      from icv83551.optim import adam

      N, D = 4, 5
      w = np.linspace(-0.4, 0.6, num=N*D).reshape(N, D)
      dw = np.linspace(-0.6, 0.4, num=N*D).reshape(N, D)
      m = np.linspace(0.6, 0.9, num=N*D).reshape(N, D)
      v = np.linspace(0.7, 0.5, num=N*D).reshape(N, D)

      config = {'learning_rate': 1e-2, 'm': m, 'v': v, 't': 5}
      next_w, _ = adam(w, dw, config=config)

      expected_next_w = np.asarray([
        [-0.40094747, -0.34836187, -0.29577703, -0.24319299, -0.19060977],
        [-0.1380274,  -0.08544591, -0.03286534,  0.01971428,  0.0722929],
        [ 0.1248705,   0.17744702,  0.23002243,  0.28259667,  0.33516969],
        [ 0.38774145,  0.44031188,  0.49288093,  0.54544852,  0.59801459]])
```

13

```
expected_v = np.asarray([
  [ 0.69966,     0.68908382,  0.67851319,  0.66794809,  0.65738853,],
  [ 0.64683452,  0.63628604,  0.6257431,   0.61520571,  0.60467385,],
  [ 0.59414753,  0.58362676,  0.57311152,  0.56260183,  0.55209767,],
  [ 0.54159906,  0.53110598,  0.52061845,  0.51013645,  0.49966,    ]])
expected_m = np.asarray([
  [ 0.48,        0.49947368,  0.51894737,  0.53842105,  0.55789474],
  [ 0.57736842,  0.59684211,  0.61631579,  0.63578947,  0.65526316],
  [ 0.67473684,  0.69421053,  0.71368421,  0.73315789,  0.75263158],
  [ 0.77210526,  0.79157895,  0.81105263,  0.83052632,  0.85       ]])

# You should see relative errors around e-7 or less
print('next_w error: ', rel_error(expected_next_w, next_w))
print('v error: ', rel_error(expected_v, config['v']))
print('m error: ', rel_error(expected_m, config['m']))
```

```
next_w error:  1.1395691798535431e-07
v error:  4.208314038113071e-09
m error:  4.214963193114416e-09
```

Once you have debugged your RMSProp and Adam implementations, run the following to train a pair of deep networks using these new update rules:

```
[12]: learning_rates = {'rmsprop': 1e-4, 'adam': 1e-3}
for update_rule in ['adam', 'rmsprop']:
    print('Running with ', update_rule)
    model = FullyConnectedNet(
        [100, 100, 100, 100, 100],
        weight_scale=5e-2
    )
    solver = Solver(
        model,
        small_data,
        num_epochs=5,
        batch_size=100,
        update_rule=update_rule,
        optim_config={'learning_rate': learning_rates[update_rule]},
        verbose=True
    )
    solvers[update_rule] = solver
    solver.train()
    print()

fig, axes = plt.subplots(3, 1, figsize=(15, 15))

axes[0].set_title('Training loss')
axes[0].set_xlabel('Iteration')
axes[1].set_title('Training accuracy')
```

```python
axes[1].set_xlabel('Epoch')
axes[2].set_title('Validation accuracy')
axes[2].set_xlabel('Epoch')

for update_rule, solver in solvers.items():
    axes[0].plot(solver.loss_history, label=f"{update_rule}")
    axes[1].plot(solver.train_acc_history, label=f"{update_rule}")
    axes[2].plot(solver.val_acc_history, label=f"{update_rule}")

for ax in axes:
    ax.legend(loc='best', ncol=4)
    ax.grid(linestyle='--', linewidth=0.5)

plt.show()
```

```
Running with  adam
(Iteration 1 / 200) loss: 3.476928
(Epoch 0 / 5) train acc: 0.126000; val_acc: 0.110000
(Iteration 11 / 200) loss: 2.027712
(Iteration 21 / 200) loss: 2.183357
(Iteration 31 / 200) loss: 1.744257
(Epoch 1 / 5) train acc: 0.363000; val_acc: 0.330000
(Iteration 41 / 200) loss: 1.707951
(Iteration 51 / 200) loss: 1.703835
(Iteration 61 / 200) loss: 2.094758
(Iteration 71 / 200) loss: 1.505558
(Epoch 2 / 5) train acc: 0.419000; val_acc: 0.362000
(Iteration 81 / 200) loss: 1.594431
(Iteration 91 / 200) loss: 1.511452
(Iteration 101 / 200) loss: 1.389230
(Iteration 111 / 200) loss: 1.465176
(Epoch 3 / 5) train acc: 0.506000; val_acc: 0.379000
(Iteration 121 / 200) loss: 1.226219
(Iteration 131 / 200) loss: 1.475588
(Iteration 141 / 200) loss: 1.397937
(Iteration 151 / 200) loss: 1.279100
(Epoch 4 / 5) train acc: 0.543000; val_acc: 0.356000
(Iteration 161 / 200) loss: 1.370917
(Iteration 171 / 200) loss: 1.291703
(Iteration 181 / 200) loss: 1.156816
(Iteration 191 / 200) loss: 1.186753
(Epoch 5 / 5) train acc: 0.595000; val_acc: 0.377000

Running with  rmsprop
(Iteration 1 / 200) loss: 2.589166
(Epoch 0 / 5) train acc: 0.119000; val_acc: 0.146000
(Iteration 11 / 200) loss: 2.032921
```

```
(Iteration 21 / 200) loss: 1.897277
(Iteration 31 / 200) loss: 1.770793
(Epoch 1 / 5) train acc: 0.381000; val_acc: 0.320000
(Iteration 41 / 200) loss: 1.895731
(Iteration 51 / 200) loss: 1.681091
(Iteration 61 / 200) loss: 1.487204
(Iteration 71 / 200) loss: 1.629973
(Epoch 2 / 5) train acc: 0.429000; val_acc: 0.350000
(Iteration 81 / 200) loss: 1.506686
(Iteration 91 / 200) loss: 1.610742
(Iteration 101 / 200) loss: 1.486124
(Iteration 111 / 200) loss: 1.559454
(Epoch 3 / 5) train acc: 0.492000; val_acc: 0.359000
(Iteration 121 / 200) loss: 1.496860
(Iteration 131 / 200) loss: 1.531552
(Iteration 141 / 200) loss: 1.550196
(Iteration 151 / 200) loss: 1.657838
(Epoch 4 / 5) train acc: 0.533000; val_acc: 0.354000
(Iteration 161 / 200) loss: 1.603105
(Iteration 171 / 200) loss: 1.405372
(Iteration 181 / 200) loss: 1.503740
(Iteration 191 / 200) loss: 1.385278
(Epoch 5 / 5) train acc: 0.531000; val_acc: 0.374000
```
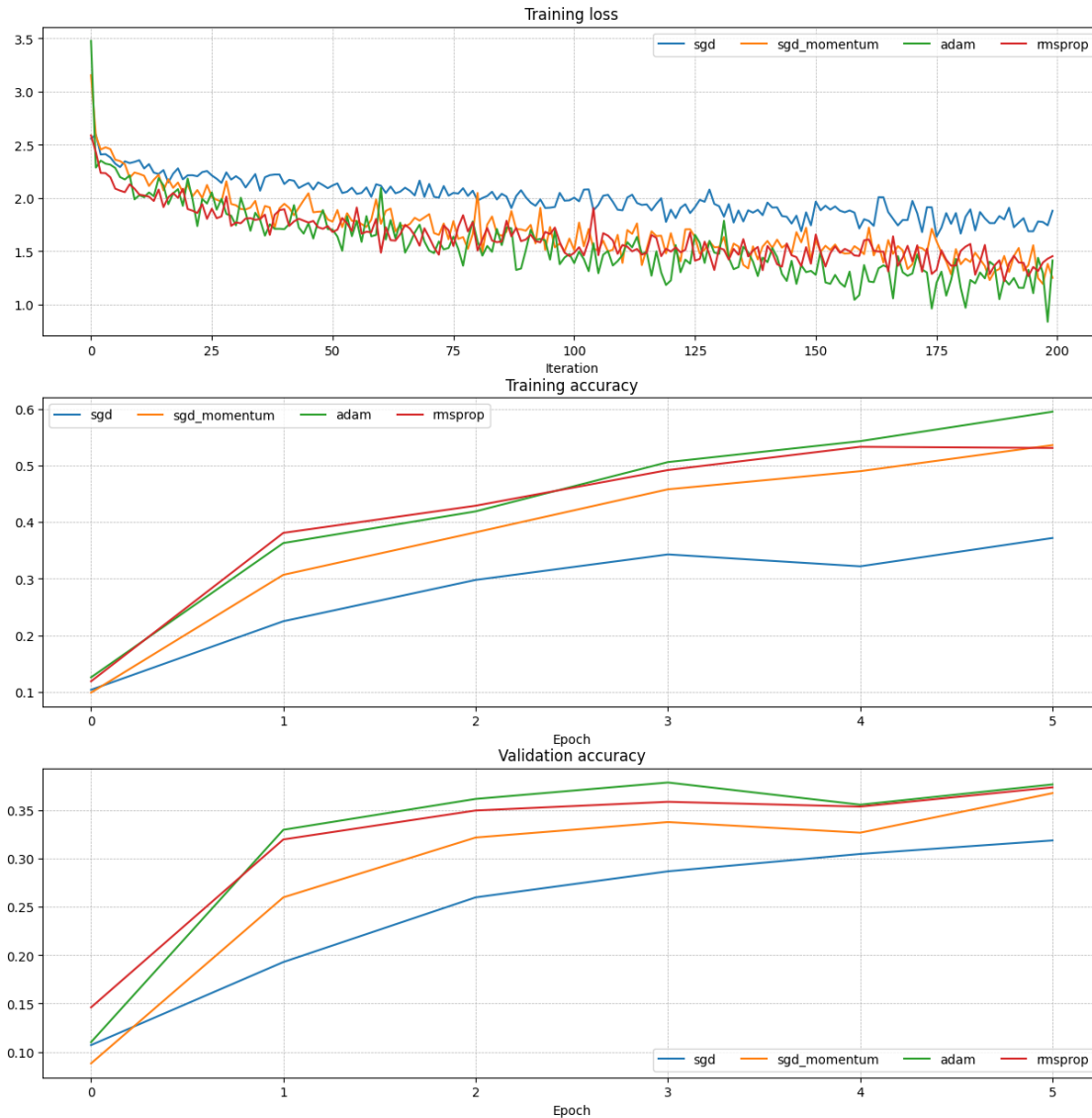
2.3 Inline Question 2:

AdaGrad, like Adam, is a per-parameter optimization method that uses the following update rule:

```
cache += dw**2
w += - learning_rate * dw / (np.sqrt(cache) + eps)
```

John notices that when he was training a network with AdaGrad that the updates became very small, and that his network was learning slowly. Using your knowledge of the AdaGrad update rule, why do you think the updates would become very small? Would Adam have the same issue?

2.4 Answer:

the cache grows over time - there is no decay, since dw**2>=0, meaning the update rule becomes smaller and smaller, meaning it will "stop" learning early. on the other hand, adam has the decay rate, for that exact reason! so it continously learns till it convirges (if possible)

3 Train a Good Model!

Train the best fully connected model that you can on CIFAR-10, storing your best model in the `best_model` variable. We require you to get at least 50% accuracy on the validation set using a fully connected network.

If you are careful it should be possible to get accuracies above 55%, but we don't require it for this part and won't assign extra credit for doing so. Later in the next assignment, we will ask you to train the best convolutional network that you can on CIFAR-10, and we would prefer that you spend your effort working on convolutional networks rather than fully connected networks.

Note: In the next assignment, you will learn techniques like BatchNormalization and Dropout which can help you train powerful models.

```python
[13]: best_model = None



    ##############################################################################
    # TODO: Train the best FullyConnectedNet that you can on CIFAR-10. You might  #
    # find batch/layer normalization and dropout useful. Store your best model in #
    # the best_model variable.                                                    #
    ##############################################################################
    # *****START OF YOUR CODE (DO NOT DELETE/MODIFY THIS LINE)*****
    highest_acc = 0
    learning_rates = {'rmsprop': 1e-4, 'adam': 1e-3, 'sgd':5e-3, 'sgd_momentum':
     ↪5e-3}
    depths = [[512], [256, 256], [128, 128, 128], [64, 64, 64, 64]]
    for update_rule, lr in learning_rates.items():
        for depth in depths:
          model = FullyConnectedNet(
              depth,
              reg=1e-5,
              weight_scale=1e-2,
          )
          solver = Solver(
              model,
              data,
              num_epochs=15,
              batch_size=200,
              update_rule=update_rule,
              optim_config={'learning_rate': learning_rates[update_rule]},
              verbose=True,
              print_every=100
```

```
        )

        solver.train()
        acc = solver.check_accuracy(data['X_val'], data['y_val'])
        if acc > highest_acc:
            highest_acc = acc
            best_model = model



    # *****END OF YOUR CODE (DO NOT DELETE/MODIFY THIS LINE)*****
    ###########################################################################
    #                          END OF YOUR CODE                               #
    ###########################################################################
```

```
(Iteration 1 / 3675) loss: 10.371776
(Epoch 0 / 15) train acc: 0.142000; val_acc: 0.135000

/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/layers.py:807:
RuntimeWarning: divide by zero encountered in log
  loss = (-np.sum(np.log(probs[np.arange(x.shape[0]), y])))/num_train

(Iteration 101 / 3675) loss: 3.228622
(Iteration 201 / 3675) loss: 2.877154
(Epoch 1 / 15) train acc: 0.451000; val_acc: 0.393000
(Iteration 301 / 3675) loss: 2.559250
(Iteration 401 / 3675) loss: 2.193378
(Epoch 2 / 15) train acc: 0.419000; val_acc: 0.407000
(Iteration 501 / 3675) loss: 1.797979
(Iteration 601 / 3675) loss: 1.993728
(Iteration 701 / 3675) loss: 1.648675
(Epoch 3 / 15) train acc: 0.538000; val_acc: 0.455000
(Iteration 801 / 3675) loss: 1.757462
(Iteration 901 / 3675) loss: 1.305498
(Epoch 4 / 15) train acc: 0.542000; val_acc: 0.430000
(Iteration 1001 / 3675) loss: 1.553760
(Iteration 1101 / 3675) loss: 1.403580
(Iteration 1201 / 3675) loss: 1.486491
(Epoch 5 / 15) train acc: 0.500000; val_acc: 0.421000
(Iteration 1301 / 3675) loss: 1.631545
(Iteration 1401 / 3675) loss: 1.943978
(Epoch 6 / 15) train acc: 0.596000; val_acc: 0.476000
(Iteration 1501 / 3675) loss: 1.289398
(Iteration 1601 / 3675) loss: 1.263158
(Iteration 1701 / 3675) loss: 1.221519
(Epoch 7 / 15) train acc: 0.619000; val_acc: 0.480000
(Iteration 1801 / 3675) loss: 1.101851
(Iteration 1901 / 3675) loss: 1.186914
(Epoch 8 / 15) train acc: 0.578000; val_acc: 0.476000
```

```
(Iteration 2001 / 3675) loss: 1.402327
(Iteration 2101 / 3675) loss: 1.033059
(Iteration 2201 / 3675) loss: 1.095445
(Epoch 9 / 15) train acc: 0.612000; val_acc: 0.479000
(Iteration 2301 / 3675) loss: 0.929011
(Iteration 2401 / 3675) loss: 1.212107
(Epoch 10 / 15) train acc: 0.589000; val_acc: 0.483000
(Iteration 2501 / 3675) loss: 1.212326
(Iteration 2601 / 3675) loss: 1.343353
(Epoch 11 / 15) train acc: 0.647000; val_acc: 0.483000
(Iteration 2701 / 3675) loss: 1.100360
(Iteration 2801 / 3675) loss: 1.107296
(Iteration 2901 / 3675) loss: 1.160280
(Epoch 12 / 15) train acc: 0.629000; val_acc: 0.480000
(Iteration 3001 / 3675) loss: 1.082813
(Iteration 3101 / 3675) loss: 1.089107
(Epoch 13 / 15) train acc: 0.634000; val_acc: 0.486000
(Iteration 3201 / 3675) loss: 0.994219
(Iteration 3301 / 3675) loss: 0.984735
(Iteration 3401 / 3675) loss: 1.046478
(Epoch 14 / 15) train acc: 0.682000; val_acc: 0.505000
(Iteration 3501 / 3675) loss: 0.981183
(Iteration 3601 / 3675) loss: 0.905250
(Epoch 15 / 15) train acc: 0.669000; val_acc: 0.485000
(Iteration 1 / 3675) loss: 2.317024
(Epoch 0 / 15) train acc: 0.169000; val_acc: 0.190000
(Iteration 101 / 3675) loss: 1.564303
(Iteration 201 / 3675) loss: 1.354884
(Epoch 1 / 15) train acc: 0.485000; val_acc: 0.460000
(Iteration 301 / 3675) loss: 1.356480
(Iteration 401 / 3675) loss: 1.362754
(Epoch 2 / 15) train acc: 0.557000; val_acc: 0.501000
(Iteration 501 / 3675) loss: 1.323457
(Iteration 601 / 3675) loss: 1.365360
(Iteration 701 / 3675) loss: 1.233010
(Epoch 3 / 15) train acc: 0.559000; val_acc: 0.513000
(Iteration 801 / 3675) loss: 1.303184
(Iteration 901 / 3675) loss: 1.068635
(Epoch 4 / 15) train acc: 0.622000; val_acc: 0.517000
(Iteration 1001 / 3675) loss: 1.195110
(Iteration 1101 / 3675) loss: 1.132555
(Iteration 1201 / 3675) loss: 1.095399
(Epoch 5 / 15) train acc: 0.640000; val_acc: 0.537000
(Iteration 1301 / 3675) loss: 1.064847
(Iteration 1401 / 3675) loss: 1.088729
(Epoch 6 / 15) train acc: 0.653000; val_acc: 0.526000
(Iteration 1501 / 3675) loss: 1.155449
(Iteration 1601 / 3675) loss: 0.858979
```

```
(Iteration 1701 / 3675) loss: 0.986198
(Epoch 7 / 15) train acc: 0.667000; val_acc: 0.534000
(Iteration 1801 / 3675) loss: 0.999483
(Iteration 1901 / 3675) loss: 0.985617
(Epoch 8 / 15) train acc: 0.686000; val_acc: 0.547000
(Iteration 2001 / 3675) loss: 0.947793
(Iteration 2101 / 3675) loss: 1.034148
(Iteration 2201 / 3675) loss: 0.871765
(Epoch 9 / 15) train acc: 0.704000; val_acc: 0.547000
(Iteration 2301 / 3675) loss: 0.851853
(Iteration 2401 / 3675) loss: 0.777271
(Epoch 10 / 15) train acc: 0.752000; val_acc: 0.563000
(Iteration 2501 / 3675) loss: 0.838486
(Iteration 2601 / 3675) loss: 0.810768
(Epoch 11 / 15) train acc: 0.730000; val_acc: 0.526000
(Iteration 2701 / 3675) loss: 0.734835
(Iteration 2801 / 3675) loss: 0.874019
(Iteration 2901 / 3675) loss: 0.711704
(Epoch 12 / 15) train acc: 0.737000; val_acc: 0.532000
(Iteration 3001 / 3675) loss: 0.754271
(Iteration 3101 / 3675) loss: 0.678475
(Epoch 13 / 15) train acc: 0.762000; val_acc: 0.560000
(Iteration 3201 / 3675) loss: 0.628834
(Iteration 3301 / 3675) loss: 0.561996
(Iteration 3401 / 3675) loss: 0.625892
(Epoch 14 / 15) train acc: 0.796000; val_acc: 0.545000
(Iteration 3501 / 3675) loss: 0.536360
(Iteration 3601 / 3675) loss: 0.550019
(Epoch 15 / 15) train acc: 0.819000; val_acc: 0.550000
(Iteration 1 / 3675) loss: 2.302696
(Epoch 0 / 15) train acc: 0.139000; val_acc: 0.142000
(Iteration 101 / 3675) loss: 1.682657
(Iteration 201 / 3675) loss: 1.716026
(Epoch 1 / 15) train acc: 0.440000; val_acc: 0.451000
(Iteration 301 / 3675) loss: 1.510737
(Iteration 401 / 3675) loss: 1.632227
(Epoch 2 / 15) train acc: 0.474000; val_acc: 0.487000
(Iteration 501 / 3675) loss: 1.416505
(Iteration 601 / 3675) loss: 1.463346
(Iteration 701 / 3675) loss: 1.362755
(Epoch 3 / 15) train acc: 0.514000; val_acc: 0.495000
(Iteration 801 / 3675) loss: 1.394408
(Iteration 901 / 3675) loss: 1.362991
(Epoch 4 / 15) train acc: 0.528000; val_acc: 0.518000
(Iteration 1001 / 3675) loss: 1.407560
(Iteration 1101 / 3675) loss: 1.422873
(Iteration 1201 / 3675) loss: 1.339101
(Epoch 5 / 15) train acc: 0.567000; val_acc: 0.518000
```

```
(Iteration 1301 / 3675) loss: 1.178352
(Iteration 1401 / 3675) loss: 1.160659
(Epoch 6 / 15) train acc: 0.582000; val_acc: 0.511000
(Iteration 1501 / 3675) loss: 1.283185
(Iteration 1601 / 3675) loss: 1.122840
(Iteration 1701 / 3675) loss: 1.345717
(Epoch 7 / 15) train acc: 0.573000; val_acc: 0.511000
(Iteration 1801 / 3675) loss: 1.209959
(Iteration 1901 / 3675) loss: 1.112214
(Epoch 8 / 15) train acc: 0.576000; val_acc: 0.522000
(Iteration 2001 / 3675) loss: 1.175946
(Iteration 2101 / 3675) loss: 1.119214
(Iteration 2201 / 3675) loss: 1.228534
(Epoch 9 / 15) train acc: 0.632000; val_acc: 0.516000
(Iteration 2301 / 3675) loss: 1.142318
(Iteration 2401 / 3675) loss: 1.130167
(Epoch 10 / 15) train acc: 0.615000; val_acc: 0.531000
(Iteration 2501 / 3675) loss: 1.155872
(Iteration 2601 / 3675) loss: 1.170959
(Epoch 11 / 15) train acc: 0.598000; val_acc: 0.521000
(Iteration 2701 / 3675) loss: 1.139503
(Iteration 2801 / 3675) loss: 1.106138
(Iteration 2901 / 3675) loss: 1.158160
(Epoch 12 / 15) train acc: 0.641000; val_acc: 0.537000
(Iteration 3001 / 3675) loss: 1.144352
(Iteration 3101 / 3675) loss: 1.037576
(Epoch 13 / 15) train acc: 0.624000; val_acc: 0.530000
(Iteration 3201 / 3675) loss: 1.131192
(Iteration 3301 / 3675) loss: 0.979744
(Iteration 3401 / 3675) loss: 1.039191
(Epoch 14 / 15) train acc: 0.624000; val_acc: 0.531000
(Iteration 3501 / 3675) loss: 1.004086
(Iteration 3601 / 3675) loss: 1.023333
(Epoch 15 / 15) train acc: 0.656000; val_acc: 0.522000
(Iteration 1 / 3675) loss: 2.302698
(Epoch 0 / 15) train acc: 0.125000; val_acc: 0.106000
(Iteration 101 / 3675) loss: 1.923740
(Iteration 201 / 3675) loss: 1.911034
(Epoch 1 / 15) train acc: 0.306000; val_acc: 0.309000
(Iteration 301 / 3675) loss: 1.742736
(Iteration 401 / 3675) loss: 1.730355
(Epoch 2 / 15) train acc: 0.332000; val_acc: 0.364000
(Iteration 501 / 3675) loss: 1.760836
(Iteration 601 / 3675) loss: 1.684510
(Iteration 701 / 3675) loss: 1.675346
(Epoch 3 / 15) train acc: 0.386000; val_acc: 0.409000
(Iteration 801 / 3675) loss: 1.538595
(Iteration 901 / 3675) loss: 1.514728
```

```
(Epoch 4 / 15) train acc: 0.403000; val_acc: 0.427000
(Iteration 1001 / 3675) loss: 1.706258
(Iteration 1101 / 3675) loss: 1.551625
(Iteration 1201 / 3675) loss: 1.563362
(Epoch 5 / 15) train acc: 0.429000; val_acc: 0.434000
(Iteration 1301 / 3675) loss: 1.485406
(Iteration 1401 / 3675) loss: 1.531802
(Epoch 6 / 15) train acc: 0.454000; val_acc: 0.457000
(Iteration 1501 / 3675) loss: 1.558383
(Iteration 1601 / 3675) loss: 1.447568
(Iteration 1701 / 3675) loss: 1.521893
(Epoch 7 / 15) train acc: 0.465000; val_acc: 0.450000
(Iteration 1801 / 3675) loss: 1.475424
(Iteration 1901 / 3675) loss: 1.524956
(Epoch 8 / 15) train acc: 0.488000; val_acc: 0.462000
(Iteration 2001 / 3675) loss: 1.464600
(Iteration 2101 / 3675) loss: 1.506039
(Iteration 2201 / 3675) loss: 1.519695
(Epoch 9 / 15) train acc: 0.479000; val_acc: 0.465000
(Iteration 2301 / 3675) loss: 1.532914
(Iteration 2401 / 3675) loss: 1.379537
(Epoch 10 / 15) train acc: 0.505000; val_acc: 0.484000
(Iteration 2501 / 3675) loss: 1.389612
(Iteration 2601 / 3675) loss: 1.449321
(Epoch 11 / 15) train acc: 0.510000; val_acc: 0.486000
(Iteration 2701 / 3675) loss: 1.443069
(Iteration 2801 / 3675) loss: 1.462107
(Iteration 2901 / 3675) loss: 1.324856
(Epoch 12 / 15) train acc: 0.503000; val_acc: 0.492000
(Iteration 3001 / 3675) loss: 1.499086
(Iteration 3101 / 3675) loss: 1.411759
(Epoch 13 / 15) train acc: 0.520000; val_acc: 0.493000
(Iteration 3201 / 3675) loss: 1.461364
(Iteration 3301 / 3675) loss: 1.544175
(Iteration 3401 / 3675) loss: 1.308635
(Epoch 14 / 15) train acc: 0.519000; val_acc: 0.492000
(Iteration 3501 / 3675) loss: 1.408069
(Iteration 3601 / 3675) loss: 1.407369
(Epoch 15 / 15) train acc: 0.526000; val_acc: 0.500000
(Iteration 1 / 3675) loss: 9.189712
(Epoch 0 / 15) train acc: 0.173000; val_acc: 0.171000
(Iteration 101 / 3675) loss: 3.507419
(Iteration 201 / 3675) loss: 1.967241
(Epoch 1 / 15) train acc: 0.434000; val_acc: 0.449000
(Iteration 301 / 3675) loss: 1.668953
(Iteration 401 / 3675) loss: 1.900196
(Epoch 2 / 15) train acc: 0.470000; val_acc: 0.447000
(Iteration 501 / 3675) loss: 2.000451
```

```
(Iteration 601 / 3675) loss: 1.607478
(Iteration 701 / 3675) loss: 1.999152
(Epoch 3 / 15) train acc: 0.390000; val_acc: 0.357000
(Iteration 801 / 3675) loss: 2.303987
(Iteration 901 / 3675) loss: 2.205779
(Epoch 4 / 15) train acc: 0.427000; val_acc: 0.427000
(Iteration 1001 / 3675) loss: 2.381907
(Iteration 1101 / 3675) loss: 1.979846
(Iteration 1201 / 3675) loss: 2.523782
(Epoch 5 / 15) train acc: 0.385000; val_acc: 0.397000
(Iteration 1301 / 3675) loss: 2.989236
(Iteration 1401 / 3675) loss: 2.276442
(Epoch 6 / 15) train acc: 0.434000; val_acc: 0.389000
(Iteration 1501 / 3675) loss: 2.766553
(Iteration 1601 / 3675) loss: 1.990901
(Iteration 1701 / 3675) loss: 3.160388
(Epoch 7 / 15) train acc: 0.454000; val_acc: 0.421000
(Iteration 1801 / 3675) loss: 2.277722
(Iteration 1901 / 3675) loss: 3.078892
(Epoch 8 / 15) train acc: 0.426000; val_acc: 0.416000
(Iteration 2001 / 3675) loss: 2.279792
(Iteration 2101 / 3675) loss: 2.829203
(Iteration 2201 / 3675) loss: 4.170337
(Epoch 9 / 15) train acc: 0.400000; val_acc: 0.344000
(Iteration 2301 / 3675) loss: inf
(Iteration 2401 / 3675) loss: 5.131044
(Epoch 10 / 15) train acc: 0.446000; val_acc: 0.406000
(Iteration 2501 / 3675) loss: 2.315143
(Iteration 2601 / 3675) loss: 2.432901
(Epoch 11 / 15) train acc: 0.489000; val_acc: 0.428000
(Iteration 2701 / 3675) loss: 2.909539
(Iteration 2801 / 3675) loss: 2.029555
(Iteration 2901 / 3675) loss: 2.365854
(Epoch 12 / 15) train acc: 0.457000; val_acc: 0.415000
(Iteration 3001 / 3675) loss: 1.701059
(Iteration 3101 / 3675) loss: 2.138751
(Epoch 13 / 15) train acc: 0.449000; val_acc: 0.417000
(Iteration 3201 / 3675) loss: 2.512920
(Iteration 3301 / 3675) loss: 2.274235
(Iteration 3401 / 3675) loss: 2.637414
(Epoch 14 / 15) train acc: 0.472000; val_acc: 0.421000
(Iteration 3501 / 3675) loss: 2.761798
(Iteration 3601 / 3675) loss: 3.375634
(Epoch 15 / 15) train acc: 0.428000; val_acc: 0.373000
(Iteration 1 / 3675) loss: 2.361338
(Epoch 0 / 15) train acc: 0.197000; val_acc: 0.187000
(Iteration 101 / 3675) loss: 1.575564
(Iteration 201 / 3675) loss: 1.730625
```

```
(Epoch 1 / 15) train acc: 0.448000; val_acc: 0.419000
(Iteration 301 / 3675) loss: 1.501871
(Iteration 401 / 3675) loss: 1.471257
(Epoch 2 / 15) train acc: 0.512000; val_acc: 0.430000
(Iteration 501 / 3675) loss: 1.397673
(Iteration 601 / 3675) loss: 1.338585
(Iteration 701 / 3675) loss: 1.433450
(Epoch 3 / 15) train acc: 0.519000; val_acc: 0.465000
(Iteration 801 / 3675) loss: 1.415856
(Iteration 901 / 3675) loss: 1.267424
(Epoch 4 / 15) train acc: 0.536000; val_acc: 0.479000
(Iteration 1001 / 3675) loss: 1.459419
(Iteration 1101 / 3675) loss: 1.341341
(Iteration 1201 / 3675) loss: 1.340973
(Epoch 5 / 15) train acc: 0.552000; val_acc: 0.521000
(Iteration 1301 / 3675) loss: 1.330814
(Iteration 1401 / 3675) loss: 1.286577
(Epoch 6 / 15) train acc: 0.531000; val_acc: 0.511000
(Iteration 1501 / 3675) loss: 1.409079
(Iteration 1601 / 3675) loss: 1.528582
(Iteration 1701 / 3675) loss: 1.241119
(Epoch 7 / 15) train acc: 0.560000; val_acc: 0.504000
(Iteration 1801 / 3675) loss: 1.260079
(Iteration 1901 / 3675) loss: 1.029138
(Epoch 8 / 15) train acc: 0.578000; val_acc: 0.507000
(Iteration 2001 / 3675) loss: 1.076848
(Iteration 2101 / 3675) loss: 1.231271
(Iteration 2201 / 3675) loss: 1.335822
(Epoch 9 / 15) train acc: 0.565000; val_acc: 0.489000
(Iteration 2301 / 3675) loss: 1.156990
(Iteration 2401 / 3675) loss: 1.077786
(Epoch 10 / 15) train acc: 0.615000; val_acc: 0.533000
(Iteration 2501 / 3675) loss: 1.229334
(Iteration 2601 / 3675) loss: 1.073363
(Epoch 11 / 15) train acc: 0.626000; val_acc: 0.523000
(Iteration 2701 / 3675) loss: 1.245150
(Iteration 2801 / 3675) loss: 1.017623
(Iteration 2901 / 3675) loss: 1.077579
(Epoch 12 / 15) train acc: 0.620000; val_acc: 0.522000
(Iteration 3001 / 3675) loss: 1.043370
(Iteration 3101 / 3675) loss: 1.135333
(Epoch 13 / 15) train acc: 0.615000; val_acc: 0.521000
(Iteration 3201 / 3675) loss: 1.065106
(Iteration 3301 / 3675) loss: 1.110399
(Iteration 3401 / 3675) loss: 1.150673
(Epoch 14 / 15) train acc: 0.667000; val_acc: 0.520000
(Iteration 3501 / 3675) loss: 1.167046
(Iteration 3601 / 3675) loss: 1.113089
```

```
(Epoch 15 / 15) train acc: 0.677000; val_acc: 0.536000
(Iteration 1 / 3675) loss: 2.303554
(Epoch 0 / 15) train acc: 0.139000; val_acc: 0.140000
(Iteration 101 / 3675) loss: 1.773344
(Iteration 201 / 3675) loss: 1.541374
(Epoch 1 / 15) train acc: 0.459000; val_acc: 0.442000
(Iteration 301 / 3675) loss: 1.535931
(Iteration 401 / 3675) loss: 1.592772
(Epoch 2 / 15) train acc: 0.505000; val_acc: 0.438000
(Iteration 501 / 3675) loss: 1.376468
(Iteration 601 / 3675) loss: 1.410845
(Iteration 701 / 3675) loss: 1.363625
(Epoch 3 / 15) train acc: 0.522000; val_acc: 0.483000
(Iteration 801 / 3675) loss: 1.405252
(Iteration 901 / 3675) loss: 1.254670
(Epoch 4 / 15) train acc: 0.559000; val_acc: 0.503000
(Iteration 1001 / 3675) loss: 1.175997
(Iteration 1101 / 3675) loss: 1.325284
(Iteration 1201 / 3675) loss: 1.158311
(Epoch 5 / 15) train acc: 0.565000; val_acc: 0.506000
(Iteration 1301 / 3675) loss: 1.172665
(Iteration 1401 / 3675) loss: 1.263638
(Epoch 6 / 15) train acc: 0.591000; val_acc: 0.523000
(Iteration 1501 / 3675) loss: 1.185042
(Iteration 1601 / 3675) loss: 1.164256
(Iteration 1701 / 3675) loss: 1.033439
(Epoch 7 / 15) train acc: 0.574000; val_acc: 0.510000
(Iteration 1801 / 3675) loss: 1.253343
(Iteration 1901 / 3675) loss: 1.180063
(Epoch 8 / 15) train acc: 0.605000; val_acc: 0.515000
(Iteration 2001 / 3675) loss: 1.172649
(Iteration 2101 / 3675) loss: 1.081204
(Iteration 2201 / 3675) loss: 1.054916
(Epoch 9 / 15) train acc: 0.641000; val_acc: 0.507000
(Iteration 2301 / 3675) loss: 0.998786
(Iteration 2401 / 3675) loss: 1.127015
(Epoch 10 / 15) train acc: 0.646000; val_acc: 0.510000
(Iteration 2501 / 3675) loss: 0.947253
(Iteration 2601 / 3675) loss: 1.048766
(Epoch 11 / 15) train acc: 0.638000; val_acc: 0.529000
(Iteration 2701 / 3675) loss: 1.127181
(Iteration 2801 / 3675) loss: 1.100135
(Iteration 2901 / 3675) loss: 0.936820
(Epoch 12 / 15) train acc: 0.678000; val_acc: 0.517000
(Iteration 3001 / 3675) loss: 1.094626
(Iteration 3101 / 3675) loss: 1.021474
(Epoch 13 / 15) train acc: 0.653000; val_acc: 0.512000
(Iteration 3201 / 3675) loss: 0.987730
```

```
(Iteration 3301 / 3675) loss: 0.810377
(Iteration 3401 / 3675) loss: 0.887916
(Epoch 14 / 15) train acc: 0.692000; val_acc: 0.507000
(Iteration 3501 / 3675) loss: 1.119810
(Iteration 3601 / 3675) loss: 0.925105
(Epoch 15 / 15) train acc: 0.661000; val_acc: 0.538000
(Iteration 1 / 3675) loss: 2.302721
(Epoch 0 / 15) train acc: 0.130000; val_acc: 0.165000
(Iteration 101 / 3675) loss: 1.820755
(Iteration 201 / 3675) loss: 1.632454
(Epoch 1 / 15) train acc: 0.398000; val_acc: 0.421000
(Iteration 301 / 3675) loss: 1.618836
(Iteration 401 / 3675) loss: 1.619452
(Epoch 2 / 15) train acc: 0.440000; val_acc: 0.462000
(Iteration 501 / 3675) loss: 1.476087
(Iteration 601 / 3675) loss: 1.432575
(Iteration 701 / 3675) loss: 1.347719
(Epoch 3 / 15) train acc: 0.518000; val_acc: 0.496000
(Iteration 801 / 3675) loss: 1.343648
(Iteration 901 / 3675) loss: 1.350035
(Epoch 4 / 15) train acc: 0.508000; val_acc: 0.498000
(Iteration 1001 / 3675) loss: 1.420807
(Iteration 1101 / 3675) loss: 1.440701
(Iteration 1201 / 3675) loss: 1.344767
(Epoch 5 / 15) train acc: 0.561000; val_acc: 0.521000
(Iteration 1301 / 3675) loss: 1.347277
(Iteration 1401 / 3675) loss: 1.311444
(Epoch 6 / 15) train acc: 0.537000; val_acc: 0.496000
(Iteration 1501 / 3675) loss: 1.217914
(Iteration 1601 / 3675) loss: 1.169498
(Iteration 1701 / 3675) loss: 1.223997
(Epoch 7 / 15) train acc: 0.594000; val_acc: 0.503000
(Iteration 1801 / 3675) loss: 1.205077
(Iteration 1901 / 3675) loss: 1.292159
(Epoch 8 / 15) train acc: 0.588000; val_acc: 0.500000
(Iteration 2001 / 3675) loss: 1.127804
(Iteration 2101 / 3675) loss: 1.203276
(Iteration 2201 / 3675) loss: 1.158380
(Epoch 9 / 15) train acc: 0.581000; val_acc: 0.504000
(Iteration 2301 / 3675) loss: 1.195017
(Iteration 2401 / 3675) loss: 1.092118
(Epoch 10 / 15) train acc: 0.605000; val_acc: 0.496000
(Iteration 2501 / 3675) loss: 1.165650
(Iteration 2601 / 3675) loss: 1.212666
(Epoch 11 / 15) train acc: 0.575000; val_acc: 0.497000
(Iteration 2701 / 3675) loss: 1.033206
(Iteration 2801 / 3675) loss: 1.079795
(Iteration 2901 / 3675) loss: 1.019366
```

```
(Epoch 12 / 15) train acc: 0.603000; val_acc: 0.495000
(Iteration 3001 / 3675) loss: 1.086406
(Iteration 3101 / 3675) loss: 1.171684
(Epoch 13 / 15) train acc: 0.615000; val_acc: 0.522000
(Iteration 3201 / 3675) loss: 1.140049
(Iteration 3301 / 3675) loss: 1.066582
(Iteration 3401 / 3675) loss: 0.964990
(Epoch 14 / 15) train acc: 0.629000; val_acc: 0.504000
(Iteration 3501 / 3675) loss: 1.134877
(Iteration 3601 / 3675) loss: 0.998059
(Epoch 15 / 15) train acc: 0.612000; val_acc: 0.501000
(Iteration 1 / 3675) loss: 9.469417
(Epoch 0 / 15) train acc: 0.132000; val_acc: 0.167000
(Iteration 101 / 3675) loss: inf

/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/layers.py:25:
RuntimeWarning: overflow encountered in matmul
  out = x.reshape(x.shape[0], -1) @ w + b
/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/layers.py:805:
RuntimeWarning: overflow encountered in subtract
  scores_prob = x - np.max(x, axis=1, keepdims=True)
/content/drive/My Drive/icv83551/assignments/assignment1/icv83551/layers.py:805:
RuntimeWarning: invalid value encountered in subtract
  scores_prob = x - np.max(x, axis=1, keepdims=True)

(Iteration 201 / 3675) loss: nan
(Epoch 1 / 15) train acc: 0.093000; val_acc: 0.087000
(Iteration 301 / 3675) loss: nan
(Iteration 401 / 3675) loss: nan
(Epoch 2 / 15) train acc: 0.104000; val_acc: 0.087000
(Iteration 501 / 3675) loss: nan
(Iteration 601 / 3675) loss: nan
(Iteration 701 / 3675) loss: nan
(Epoch 3 / 15) train acc: 0.094000; val_acc: 0.087000
(Iteration 801 / 3675) loss: nan
(Iteration 901 / 3675) loss: nan
(Epoch 4 / 15) train acc: 0.112000; val_acc: 0.087000
(Iteration 1001 / 3675) loss: nan
(Iteration 1101 / 3675) loss: nan
(Iteration 1201 / 3675) loss: nan
(Epoch 5 / 15) train acc: 0.098000; val_acc: 0.087000
(Iteration 1301 / 3675) loss: nan
(Iteration 1401 / 3675) loss: nan
(Epoch 6 / 15) train acc: 0.108000; val_acc: 0.087000
(Iteration 1501 / 3675) loss: nan
(Iteration 1601 / 3675) loss: nan
(Iteration 1701 / 3675) loss: nan
(Epoch 7 / 15) train acc: 0.108000; val_acc: 0.087000
(Iteration 1801 / 3675) loss: nan
```

```
(Iteration 1901 / 3675) loss: nan
(Epoch 8 / 15) train acc: 0.102000; val_acc: 0.087000
(Iteration 2001 / 3675) loss: nan
(Iteration 2101 / 3675) loss: nan
(Iteration 2201 / 3675) loss: nan
(Epoch 9 / 15) train acc: 0.095000; val_acc: 0.087000
(Iteration 2301 / 3675) loss: nan
(Iteration 2401 / 3675) loss: nan
(Epoch 10 / 15) train acc: 0.107000; val_acc: 0.087000
(Iteration 2501 / 3675) loss: nan
(Iteration 2601 / 3675) loss: nan
(Epoch 11 / 15) train acc: 0.128000; val_acc: 0.087000
(Iteration 2701 / 3675) loss: nan
(Iteration 2801 / 3675) loss: nan
(Iteration 2901 / 3675) loss: nan
(Epoch 12 / 15) train acc: 0.094000; val_acc: 0.087000
(Iteration 3001 / 3675) loss: nan
(Iteration 3101 / 3675) loss: nan
(Epoch 13 / 15) train acc: 0.098000; val_acc: 0.087000
(Iteration 3201 / 3675) loss: nan
(Iteration 3301 / 3675) loss: nan
(Iteration 3401 / 3675) loss: nan
(Epoch 14 / 15) train acc: 0.105000; val_acc: 0.087000
(Iteration 3501 / 3675) loss: nan
(Iteration 3601 / 3675) loss: nan
(Epoch 15 / 15) train acc: 0.100000; val_acc: 0.087000
(Iteration 1 / 3675) loss: 2.375068
(Epoch 0 / 15) train acc: 0.184000; val_acc: 0.187000
(Iteration 101 / 3675) loss: 1.591674
(Iteration 201 / 3675) loss: 1.590574
(Epoch 1 / 15) train acc: 0.433000; val_acc: 0.438000
(Iteration 301 / 3675) loss: 1.699482
(Iteration 401 / 3675) loss: 1.464154
(Epoch 2 / 15) train acc: 0.503000; val_acc: 0.450000
(Iteration 501 / 3675) loss: 1.443437
(Iteration 601 / 3675) loss: 1.285249
(Iteration 701 / 3675) loss: 1.282679
(Epoch 3 / 15) train acc: 0.533000; val_acc: 0.489000
(Iteration 801 / 3675) loss: 1.417191
(Iteration 901 / 3675) loss: 1.393182
(Epoch 4 / 15) train acc: 0.554000; val_acc: 0.492000
(Iteration 1001 / 3675) loss: 1.296211
(Iteration 1101 / 3675) loss: 1.213178
(Iteration 1201 / 3675) loss: 1.205603
(Epoch 5 / 15) train acc: 0.594000; val_acc: 0.508000
(Iteration 1301 / 3675) loss: 1.113539
(Iteration 1401 / 3675) loss: 1.291247
(Epoch 6 / 15) train acc: 0.592000; val_acc: 0.526000
```

```
(Iteration 1501 / 3675) loss: 1.229640
(Iteration 1601 / 3675) loss: 1.046004
(Iteration 1701 / 3675) loss: 1.250506
(Epoch 7 / 15) train acc: 0.615000; val_acc: 0.524000
(Iteration 1801 / 3675) loss: 1.156990
(Iteration 1901 / 3675) loss: 1.114680
(Epoch 8 / 15) train acc: 0.647000; val_acc: 0.525000
(Iteration 2001 / 3675) loss: 1.165553
(Iteration 2101 / 3675) loss: 1.014973
(Iteration 2201 / 3675) loss: 0.907843
(Epoch 9 / 15) train acc: 0.642000; val_acc: 0.523000
(Iteration 2301 / 3675) loss: 1.124281
(Iteration 2401 / 3675) loss: 1.053890
(Epoch 10 / 15) train acc: 0.708000; val_acc: 0.536000
(Iteration 2501 / 3675) loss: 0.998742
(Iteration 2601 / 3675) loss: 1.006271
(Epoch 11 / 15) train acc: 0.702000; val_acc: 0.509000
(Iteration 2701 / 3675) loss: 0.913224
(Iteration 2801 / 3675) loss: 0.805975
(Iteration 2901 / 3675) loss: 0.847626
(Epoch 12 / 15) train acc: 0.728000; val_acc: 0.517000
(Iteration 3001 / 3675) loss: 0.774833
(Iteration 3101 / 3675) loss: 0.953245
(Epoch 13 / 15) train acc: 0.706000; val_acc: 0.530000
(Iteration 3201 / 3675) loss: 0.873622
(Iteration 3301 / 3675) loss: 0.866532
(Iteration 3401 / 3675) loss: 0.710046
(Epoch 14 / 15) train acc: 0.754000; val_acc: 0.517000
(Iteration 3501 / 3675) loss: 0.891441
(Iteration 3601 / 3675) loss: 0.766569
(Epoch 15 / 15) train acc: 0.775000; val_acc: 0.519000
(Iteration 1 / 3675) loss: 2.303508
(Epoch 0 / 15) train acc: 0.107000; val_acc: 0.101000
(Iteration 101 / 3675) loss: 2.281242
(Iteration 201 / 3675) loss: 2.083677
(Epoch 1 / 15) train acc: 0.253000; val_acc: 0.271000
(Iteration 301 / 3675) loss: 1.980783
(Iteration 401 / 3675) loss: 1.909021
(Epoch 2 / 15) train acc: 0.327000; val_acc: 0.355000
(Iteration 501 / 3675) loss: 1.839034
(Iteration 601 / 3675) loss: 1.731875
(Iteration 701 / 3675) loss: 1.680105
(Epoch 3 / 15) train acc: 0.407000; val_acc: 0.403000
(Iteration 801 / 3675) loss: 1.538453
(Iteration 901 / 3675) loss: 1.674546
(Epoch 4 / 15) train acc: 0.448000; val_acc: 0.440000
(Iteration 1001 / 3675) loss: 1.699985
(Iteration 1101 / 3675) loss: 1.555948
```

```
(Iteration 1201 / 3675) loss: 1.594326
(Epoch 5 / 15) train acc: 0.426000; val_acc: 0.468000
(Iteration 1301 / 3675) loss: 1.465993
(Iteration 1401 / 3675) loss: 1.498003
(Epoch 6 / 15) train acc: 0.512000; val_acc: 0.476000
(Iteration 1501 / 3675) loss: 1.407056
(Iteration 1601 / 3675) loss: 1.466923
(Iteration 1701 / 3675) loss: 1.438884
(Epoch 7 / 15) train acc: 0.529000; val_acc: 0.493000
(Iteration 1801 / 3675) loss: 1.412980
(Iteration 1901 / 3675) loss: 1.431027
(Epoch 8 / 15) train acc: 0.499000; val_acc: 0.485000
(Iteration 2001 / 3675) loss: 1.301828
(Iteration 2101 / 3675) loss: 1.230399
(Iteration 2201 / 3675) loss: 1.366100
(Epoch 9 / 15) train acc: 0.498000; val_acc: 0.512000
(Iteration 2301 / 3675) loss: 1.246310
(Iteration 2401 / 3675) loss: 1.297798
(Epoch 10 / 15) train acc: 0.527000; val_acc: 0.495000
(Iteration 2501 / 3675) loss: 1.199770
(Iteration 2601 / 3675) loss: 1.278391
(Epoch 11 / 15) train acc: 0.567000; val_acc: 0.518000
(Iteration 2701 / 3675) loss: 1.427462
(Iteration 2801 / 3675) loss: 1.250011
(Iteration 2901 / 3675) loss: 1.410253
(Epoch 12 / 15) train acc: 0.530000; val_acc: 0.506000
(Iteration 3001 / 3675) loss: 1.111148
(Iteration 3101 / 3675) loss: 1.386864
(Epoch 13 / 15) train acc: 0.577000; val_acc: 0.519000
(Iteration 3201 / 3675) loss: 1.163781
(Iteration 3301 / 3675) loss: 1.291846
(Iteration 3401 / 3675) loss: 1.111462
(Epoch 14 / 15) train acc: 0.582000; val_acc: 0.519000
(Iteration 3501 / 3675) loss: 1.113570
(Iteration 3601 / 3675) loss: 1.260462
(Epoch 15 / 15) train acc: 0.566000; val_acc: 0.521000
(Iteration 1 / 3675) loss: 2.302682
(Epoch 0 / 15) train acc: 0.085000; val_acc: 0.098000
(Iteration 101 / 3675) loss: 2.302583
(Iteration 201 / 3675) loss: 2.302738
(Epoch 1 / 15) train acc: 0.096000; val_acc: 0.098000
(Iteration 301 / 3675) loss: 2.302683
(Iteration 401 / 3675) loss: 2.302725
(Epoch 2 / 15) train acc: 0.107000; val_acc: 0.104000
(Iteration 501 / 3675) loss: 2.302912
(Iteration 601 / 3675) loss: 2.302484
(Iteration 701 / 3675) loss: 2.302587
(Epoch 3 / 15) train acc: 0.090000; val_acc: 0.099000
```

```
(Iteration 801 / 3675) loss: 2.302882
(Iteration 901 / 3675) loss: 2.302670
(Epoch 4 / 15) train acc: 0.109000; val_acc: 0.097000
(Iteration 1001 / 3675) loss: 2.302840
(Iteration 1101 / 3675) loss: 2.302632
(Iteration 1201 / 3675) loss: 2.302724
(Epoch 5 / 15) train acc: 0.108000; val_acc: 0.098000
(Iteration 1301 / 3675) loss: 2.302490
(Iteration 1401 / 3675) loss: 2.302270
(Epoch 6 / 15) train acc: 0.095000; val_acc: 0.098000
(Iteration 1501 / 3675) loss: 2.302586
(Iteration 1601 / 3675) loss: 2.302454
(Iteration 1701 / 3675) loss: 2.302744
(Epoch 7 / 15) train acc: 0.101000; val_acc: 0.098000
(Iteration 1801 / 3675) loss: 2.302258
(Iteration 1901 / 3675) loss: 2.301950
(Epoch 8 / 15) train acc: 0.092000; val_acc: 0.098000
(Iteration 2001 / 3675) loss: 2.301914
(Iteration 2101 / 3675) loss: 2.302280
(Iteration 2201 / 3675) loss: 2.302690
(Epoch 9 / 15) train acc: 0.097000; val_acc: 0.098000
(Iteration 2301 / 3675) loss: 2.301987
(Iteration 2401 / 3675) loss: 2.302868
(Epoch 10 / 15) train acc: 0.107000; val_acc: 0.105000
(Iteration 2501 / 3675) loss: 2.302367
(Iteration 2601 / 3675) loss: 2.302527
(Epoch 11 / 15) train acc: 0.124000; val_acc: 0.101000
(Iteration 2701 / 3675) loss: 2.302356
(Iteration 2801 / 3675) loss: 2.302185
(Iteration 2901 / 3675) loss: 2.302519
(Epoch 12 / 15) train acc: 0.157000; val_acc: 0.138000
(Iteration 3001 / 3675) loss: 2.302355
(Iteration 3101 / 3675) loss: 2.302367
(Epoch 13 / 15) train acc: 0.132000; val_acc: 0.101000
(Iteration 3201 / 3675) loss: 2.302604
(Iteration 3301 / 3675) loss: 2.301974
(Iteration 3401 / 3675) loss: 2.302290
(Epoch 14 / 15) train acc: 0.119000; val_acc: 0.106000
(Iteration 3501 / 3675) loss: 2.302192
(Iteration 3601 / 3675) loss: 2.301990
(Epoch 15 / 15) train acc: 0.114000; val_acc: 0.088000
(Iteration 1 / 3675) loss: 8.183618
(Epoch 0 / 15) train acc: 0.128000; val_acc: 0.144000
(Iteration 101 / 3675) loss: nan
(Iteration 201 / 3675) loss: nan
(Epoch 1 / 15) train acc: 0.108000; val_acc: 0.087000
(Iteration 301 / 3675) loss: nan
(Iteration 401 / 3675) loss: nan
```

```
(Epoch 2 / 15) train acc: 0.114000; val_acc: 0.087000
(Iteration 501 / 3675) loss: nan
(Iteration 601 / 3675) loss: nan
(Iteration 701 / 3675) loss: nan
(Epoch 3 / 15) train acc: 0.089000; val_acc: 0.087000
(Iteration 801 / 3675) loss: nan
(Iteration 901 / 3675) loss: nan
(Epoch 4 / 15) train acc: 0.091000; val_acc: 0.087000
(Iteration 1001 / 3675) loss: nan
(Iteration 1101 / 3675) loss: nan
(Iteration 1201 / 3675) loss: nan
(Epoch 5 / 15) train acc: 0.118000; val_acc: 0.087000
(Iteration 1301 / 3675) loss: nan
(Iteration 1401 / 3675) loss: nan
(Epoch 6 / 15) train acc: 0.100000; val_acc: 0.087000
(Iteration 1501 / 3675) loss: nan
(Iteration 1601 / 3675) loss: nan
(Iteration 1701 / 3675) loss: nan
(Epoch 7 / 15) train acc: 0.100000; val_acc: 0.087000
(Iteration 1801 / 3675) loss: nan
(Iteration 1901 / 3675) loss: nan
(Epoch 8 / 15) train acc: 0.101000; val_acc: 0.087000
(Iteration 2001 / 3675) loss: nan
(Iteration 2101 / 3675) loss: nan
(Iteration 2201 / 3675) loss: nan
(Epoch 9 / 15) train acc: 0.094000; val_acc: 0.087000
(Iteration 2301 / 3675) loss: nan
(Iteration 2401 / 3675) loss: nan
(Epoch 10 / 15) train acc: 0.093000; val_acc: 0.087000
(Iteration 2501 / 3675) loss: nan
(Iteration 2601 / 3675) loss: nan
(Epoch 11 / 15) train acc: 0.110000; val_acc: 0.087000
(Iteration 2701 / 3675) loss: nan
(Iteration 2801 / 3675) loss: nan
(Iteration 2901 / 3675) loss: nan
(Epoch 12 / 15) train acc: 0.105000; val_acc: 0.087000
(Iteration 3001 / 3675) loss: nan
(Iteration 3101 / 3675) loss: nan
(Epoch 13 / 15) train acc: 0.098000; val_acc: 0.087000
(Iteration 3201 / 3675) loss: nan
(Iteration 3301 / 3675) loss: nan
(Iteration 3401 / 3675) loss: nan
(Epoch 14 / 15) train acc: 0.105000; val_acc: 0.087000
(Iteration 3501 / 3675) loss: nan
(Iteration 3601 / 3675) loss: nan
(Epoch 15 / 15) train acc: 0.085000; val_acc: 0.087000
(Iteration 1 / 3675) loss: 2.423515
(Epoch 0 / 15) train acc: 0.136000; val_acc: 0.137000
```

```
(Iteration 101 / 3675) loss: 1.736534
(Iteration 201 / 3675) loss: 1.711453
(Epoch 1 / 15) train acc: 0.415000; val_acc: 0.395000
(Iteration 301 / 3675) loss: 1.707605
(Iteration 401 / 3675) loss: 1.625291
(Epoch 2 / 15) train acc: 0.402000; val_acc: 0.423000
(Iteration 501 / 3675) loss: 1.583847
(Iteration 601 / 3675) loss: 1.571706
(Iteration 701 / 3675) loss: 1.731502
(Epoch 3 / 15) train acc: 0.440000; val_acc: 0.404000
(Iteration 801 / 3675) loss: 1.589360
(Iteration 901 / 3675) loss: 1.609557
(Epoch 4 / 15) train acc: 0.437000; val_acc: 0.390000
(Iteration 1001 / 3675) loss: 1.631195
(Iteration 1101 / 3675) loss: 1.557567
(Iteration 1201 / 3675) loss: 1.616090
(Epoch 5 / 15) train acc: 0.441000; val_acc: 0.430000
(Iteration 1301 / 3675) loss: 1.700120
(Iteration 1401 / 3675) loss: 1.577748
(Epoch 6 / 15) train acc: 0.473000; val_acc: 0.432000
(Iteration 1501 / 3675) loss: 1.570938
(Iteration 1601 / 3675) loss: 1.670768
(Iteration 1701 / 3675) loss: 1.550140
(Epoch 7 / 15) train acc: 0.446000; val_acc: 0.394000
(Iteration 1801 / 3675) loss: 1.482187
(Iteration 1901 / 3675) loss: 1.482800
(Epoch 8 / 15) train acc: 0.477000; val_acc: 0.406000
(Iteration 2001 / 3675) loss: 1.467665
(Iteration 2101 / 3675) loss: 1.586466
(Iteration 2201 / 3675) loss: 1.368340
(Epoch 9 / 15) train acc: 0.462000; val_acc: 0.406000
(Iteration 2301 / 3675) loss: 1.543861
(Iteration 2401 / 3675) loss: 1.680315
(Epoch 10 / 15) train acc: 0.352000; val_acc: 0.356000
(Iteration 2501 / 3675) loss: 1.586913
(Iteration 2601 / 3675) loss: 1.581729
(Epoch 11 / 15) train acc: 0.432000; val_acc: 0.411000
(Iteration 2701 / 3675) loss: 1.473680
(Iteration 2801 / 3675) loss: 1.695848
(Iteration 2901 / 3675) loss: 1.712490
(Epoch 12 / 15) train acc: 0.456000; val_acc: 0.409000
(Iteration 3001 / 3675) loss: 1.671621
(Iteration 3101 / 3675) loss: 1.617712
(Epoch 13 / 15) train acc: 0.410000; val_acc: 0.367000
(Iteration 3201 / 3675) loss: 1.628070
(Iteration 3301 / 3675) loss: 1.603826
(Iteration 3401 / 3675) loss: 1.579085
(Epoch 14 / 15) train acc: 0.439000; val_acc: 0.392000
```

```
(Iteration 3501 / 3675) loss: 1.735787
(Iteration 3601 / 3675) loss: 1.937650
(Epoch 15 / 15) train acc: 0.433000; val_acc: 0.372000
(Iteration 1 / 3675) loss: 2.303576
(Epoch 0 / 15) train acc: 0.115000; val_acc: 0.091000
(Iteration 101 / 3675) loss: 1.749427
(Iteration 201 / 3675) loss: 1.606856
(Epoch 1 / 15) train acc: 0.405000; val_acc: 0.436000
(Iteration 301 / 3675) loss: 1.532796
(Iteration 401 / 3675) loss: 1.488497
(Epoch 2 / 15) train acc: 0.505000; val_acc: 0.472000
(Iteration 501 / 3675) loss: 1.511083
(Iteration 601 / 3675) loss: 1.495944
(Iteration 701 / 3675) loss: 1.364084
(Epoch 3 / 15) train acc: 0.514000; val_acc: 0.485000
(Iteration 801 / 3675) loss: 1.250389
(Iteration 901 / 3675) loss: 1.460163
(Epoch 4 / 15) train acc: 0.542000; val_acc: 0.495000
(Iteration 1001 / 3675) loss: 1.125401
(Iteration 1101 / 3675) loss: 1.288700
(Iteration 1201 / 3675) loss: 1.317034
(Epoch 5 / 15) train acc: 0.543000; val_acc: 0.482000
(Iteration 1301 / 3675) loss: 1.284253
(Iteration 1401 / 3675) loss: 1.155791
(Epoch 6 / 15) train acc: 0.561000; val_acc: 0.529000
(Iteration 1501 / 3675) loss: 1.204025
(Iteration 1601 / 3675) loss: 1.238924
(Iteration 1701 / 3675) loss: 1.074323
(Epoch 7 / 15) train acc: 0.609000; val_acc: 0.526000
(Iteration 1801 / 3675) loss: 1.017562
(Iteration 1901 / 3675) loss: 1.081714
(Epoch 8 / 15) train acc: 0.625000; val_acc: 0.532000
(Iteration 2001 / 3675) loss: 1.098002
(Iteration 2101 / 3675) loss: 0.917812
(Iteration 2201 / 3675) loss: 1.099278
(Epoch 9 / 15) train acc: 0.641000; val_acc: 0.508000
(Iteration 2301 / 3675) loss: 1.032163
(Iteration 2401 / 3675) loss: 0.896366
(Epoch 10 / 15) train acc: 0.653000; val_acc: 0.547000
(Iteration 2501 / 3675) loss: 1.032947
(Iteration 2601 / 3675) loss: 1.014274
(Epoch 11 / 15) train acc: 0.630000; val_acc: 0.531000
(Iteration 2701 / 3675) loss: 1.039456
(Iteration 2801 / 3675) loss: 0.920657
(Iteration 2901 / 3675) loss: 1.036338
(Epoch 12 / 15) train acc: 0.676000; val_acc: 0.530000
(Iteration 3001 / 3675) loss: 1.000898
(Iteration 3101 / 3675) loss: 0.973728
```

```
(Epoch 13 / 15) train acc: 0.635000; val_acc: 0.515000
(Iteration 3201 / 3675) loss: 1.127521
(Iteration 3301 / 3675) loss: 0.891173
(Iteration 3401 / 3675) loss: 0.938963
(Epoch 14 / 15) train acc: 0.637000; val_acc: 0.488000
(Iteration 3501 / 3675) loss: 0.898585
(Iteration 3601 / 3675) loss: 0.939027
(Epoch 15 / 15) train acc: 0.671000; val_acc: 0.515000
(Iteration 1 / 3675) loss: 2.302672
(Epoch 0 / 15) train acc: 0.106000; val_acc: 0.091000
(Iteration 101 / 3675) loss: 2.302108
(Iteration 201 / 3675) loss: 2.302447
(Epoch 1 / 15) train acc: 0.094000; val_acc: 0.087000
(Iteration 301 / 3675) loss: 2.301656
(Iteration 401 / 3675) loss: 2.301052
(Epoch 2 / 15) train acc: 0.164000; val_acc: 0.153000
(Iteration 501 / 3675) loss: 2.162513
(Iteration 601 / 3675) loss: 1.948777
(Iteration 701 / 3675) loss: 1.817679
(Epoch 3 / 15) train acc: 0.327000; val_acc: 0.290000
(Iteration 801 / 3675) loss: 1.838476
(Iteration 901 / 3675) loss: 1.628007
(Epoch 4 / 15) train acc: 0.391000; val_acc: 0.397000
(Iteration 1001 / 3675) loss: 1.553056
(Iteration 1101 / 3675) loss: 1.578937
(Iteration 1201 / 3675) loss: 1.497257
(Epoch 5 / 15) train acc: 0.451000; val_acc: 0.454000
(Iteration 1301 / 3675) loss: 1.468449
(Iteration 1401 / 3675) loss: 1.449315
(Epoch 6 / 15) train acc: 0.463000; val_acc: 0.463000
(Iteration 1501 / 3675) loss: 1.483363
(Iteration 1601 / 3675) loss: 1.492754
(Iteration 1701 / 3675) loss: 1.381799
(Epoch 7 / 15) train acc: 0.510000; val_acc: 0.463000
(Iteration 1801 / 3675) loss: 1.362858
(Iteration 1901 / 3675) loss: 1.251240
(Epoch 8 / 15) train acc: 0.490000; val_acc: 0.495000
(Iteration 2001 / 3675) loss: 1.316838
(Iteration 2101 / 3675) loss: 1.379205
(Iteration 2201 / 3675) loss: 1.446030
(Epoch 9 / 15) train acc: 0.541000; val_acc: 0.490000
(Iteration 2301 / 3675) loss: 1.297366
(Iteration 2401 / 3675) loss: 1.327500
(Epoch 10 / 15) train acc: 0.555000; val_acc: 0.506000
(Iteration 2501 / 3675) loss: 1.087059
(Iteration 2601 / 3675) loss: 1.358631
(Epoch 11 / 15) train acc: 0.571000; val_acc: 0.509000
(Iteration 2701 / 3675) loss: 1.221719
```

```
(Iteration 2801 / 3675) loss: 1.214519
(Iteration 2901 / 3675) loss: 1.173170
(Epoch 12 / 15) train acc: 0.574000; val_acc: 0.508000
(Iteration 3001 / 3675) loss: 1.221797
(Iteration 3101 / 3675) loss: 1.240206
(Epoch 13 / 15) train acc: 0.599000; val_acc: 0.510000
(Iteration 3201 / 3675) loss: 1.225260
(Iteration 3301 / 3675) loss: 1.235365
(Iteration 3401 / 3675) loss: 1.162702
(Epoch 14 / 15) train acc: 0.575000; val_acc: 0.488000
(Iteration 3501 / 3675) loss: 1.223515
(Iteration 3601 / 3675) loss: 1.188628
(Epoch 15 / 15) train acc: 0.583000; val_acc: 0.510000
```

4 Test Your Model!

Run your best model on the validation and test sets. You should achieve at least 50% accuracy on the validation set and the test set.

```python
[14]: y_test_pred = np.argmax(best_model.loss(data['X_test']), axis=1)
      y_val_pred = np.argmax(best_model.loss(data['X_val']), axis=1)
      print('Validation set accuracy: ', (y_val_pred == data['y_val']).mean())
      print('Test set accuracy: ', (y_test_pred == data['y_test']).mean())
```

```
Validation set accuracy:  0.563
Test set accuracy:  0.548
```