Git Good

A Deeper Look at Generating Better Commit Messages

Group 47: Aman Dhar, Jackson Leisure, Riku Miyao, Matthew Sit

Problem Statement

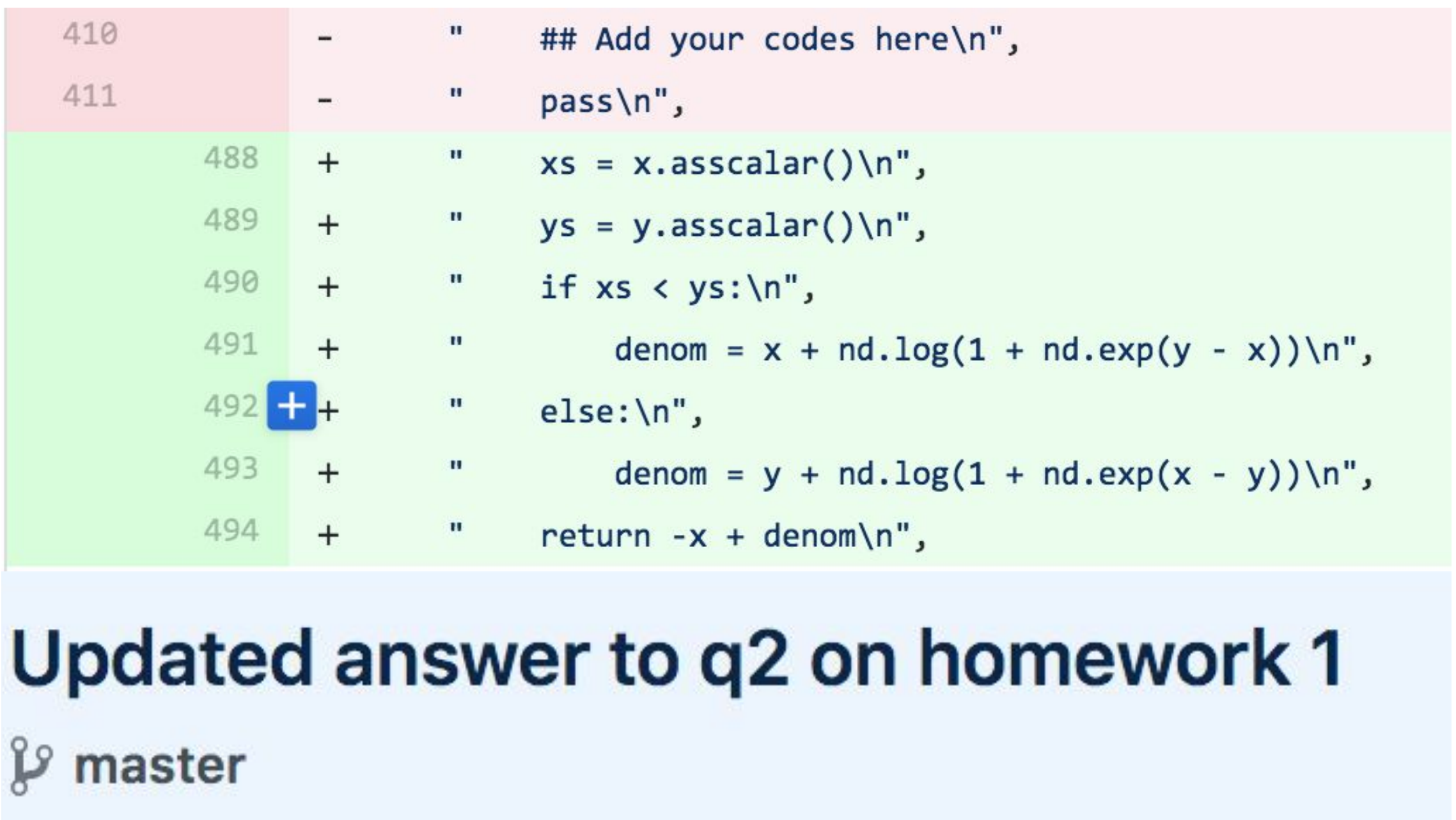
git

Many large-scale software projects make use of git, one of the most popular version control systems. When using git, users are able to describe the changes they make to a codebase at each step. At UC Berkeley, git is generally introduced to computer science students in CS 61B, but it can be difficult for first-time users to know what descriptions are the "best" when using the software.

Our goal is to use professional git repositories to train a model which can eventually assist Cal students in determining the "goodness" of their commit descriptions, in order to allow them to use git more effectively.

Background

- Git is a distributed **version control system** that allows multiple users to collaborate on software development projects
- Once a user has completed some changes to a codebase, they can save those changes to the repository by creating a **commit** with those changes
- Commits store **diff** information, showing the lines of code that were updated



```
-    "    ## Add your codes here\n",
-    "    pass\n",
+    "    xs = x.asscalar()\n",
+    "    ys = y.asscalar()\n",
+    "    if xs < ys:\n",
+    "        denom = x + nd.log(1 + nd.exp(y - x))\n",
+    "    else:\n",
+    "        denom = y + nd.log(1 + nd.exp(x - y))\n",
+    "    return -x + denom\n",
```

Updated answer to q2 on homework 1

master

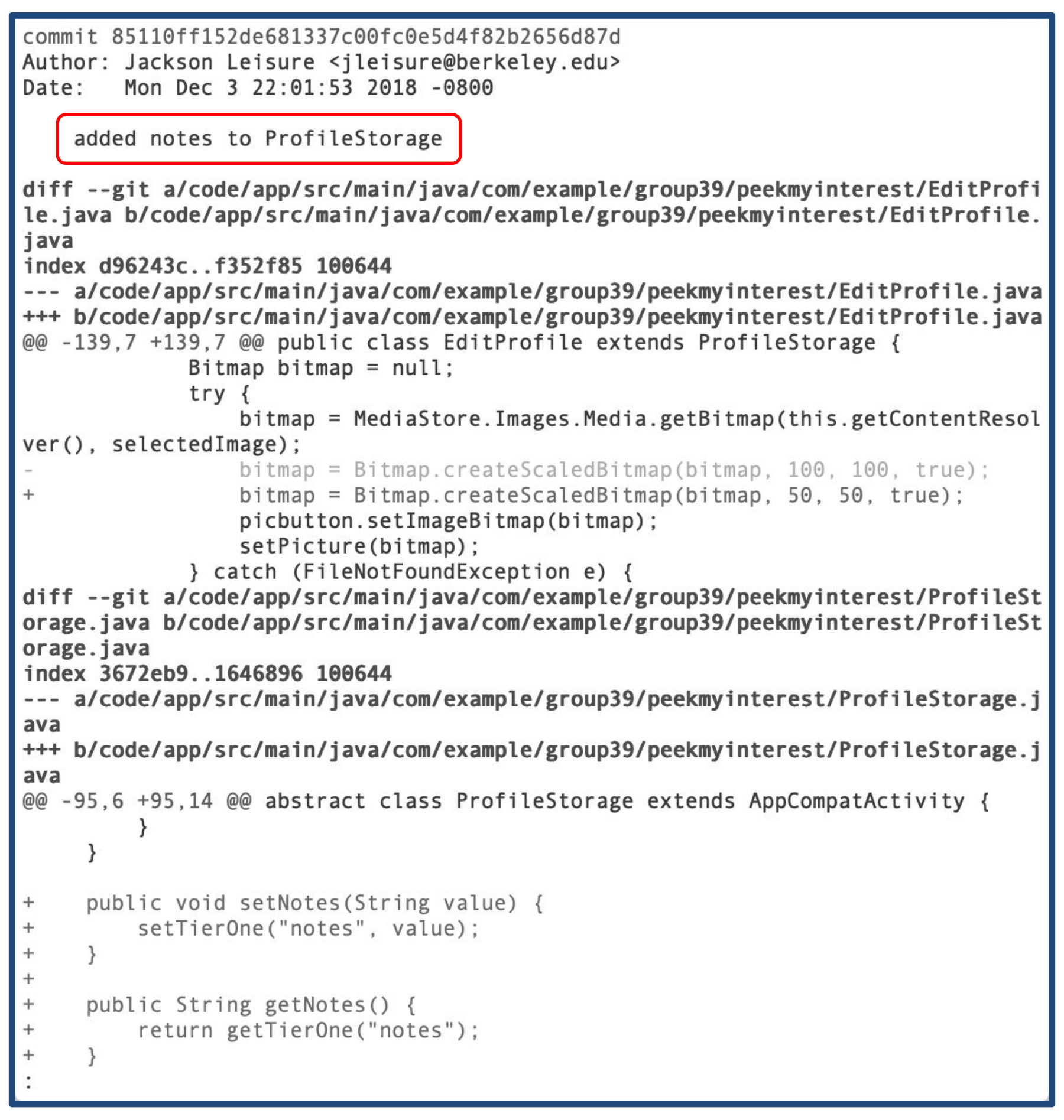
Figure 1: An example of a diff (upper) with its associated commit message (lower). The red shaded area highlights lines that were removed, while the green shaded area highlights lines that were added.

- Each commit has an associated **commit message** that summarizes the changes introduced in the commit

Data Cleaning

- Filtered out some common commit messages that deal with special cases, as these will not be as helpful to predict for students
  - For example, merges generally have a default message generated by git ("Merge branch x of project" or "Merge pull request #x from project", etc.)
- Need to use samples of the data due to memory constraints
- For language model, used vocabulary of size 20,000, with <start>, <pad>, and <unk> tokens included

Descriptive Commit Message



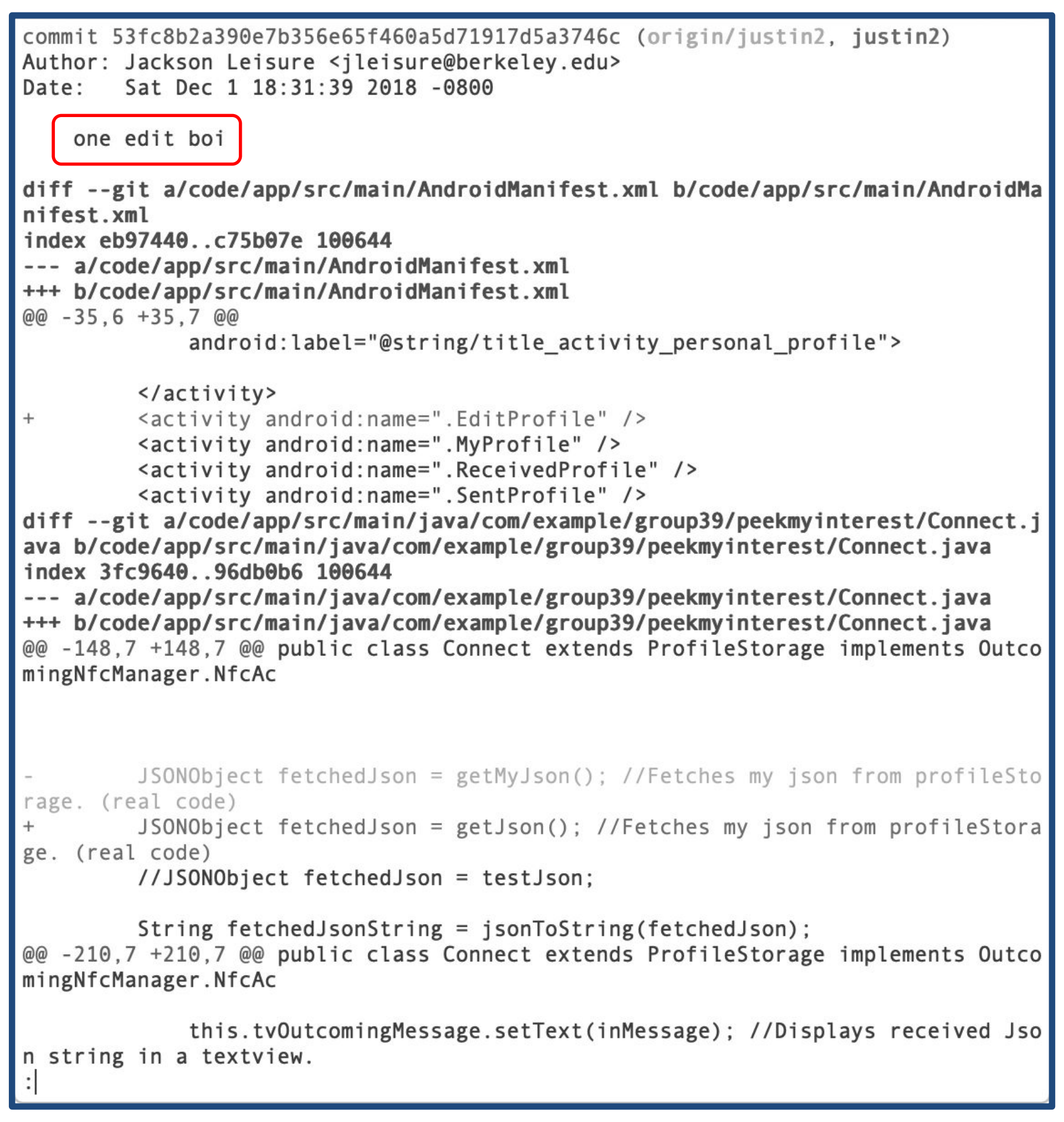
```
commit 85110ff152de681337c00fc0e5d4f82b2656d87d
Author: Jackson Leisure <jleisure@berkeley.edu>
Date:   Mon Dec 3 22:01:53 2018 -0800

    added notes to ProfileStorage

diff --git a/code/app/src/main/java/com/example/group39/peekmyinterest/EditProfi
le.java b/code/app/src/main/java/com/example/group39/peekmyinterest/EditProfile.
java
index d96243c..f352f85 100644
--- a/code/app/src/main/java/com/example/group39/peekmyinterest/EditProfile.java
+++ b/code/app/src/main/java/com/example/group39/peekmyinterest/EditProfile.java
@@ -139,7 +139,7 @@ public class EditProfile extends ProfileStorage {
             Bitmap bitmap = null;
             try {
                 bitmap = MediaStore.Images.Media.getBitmap(this.getContentResol
ver(), selectedImage);
-                bitmap = Bitmap.createScaledBitmap(bitmap, 100, 100, true);
+                bitmap = Bitmap.createScaledBitmap(bitmap, 50, 50, true);
                 picbutton.setImageBitmap(bitmap);
                 setPicture(bitmap);
             } catch (FileNotFoundException e) {
diff --git a/code/app/src/main/java/com/example/group39/peekmyinterest/ProfileSt
orage.java b/code/app/src/main/java/com/example/group39/peekmyinterest/ProfileSt
orage.java
index 3672eb9..1646896 100644
--- a/code/app/src/main/java/com/example/group39/peekmyinterest/ProfileStorage.j
ava
+++ b/code/app/src/main/java/com/example/group39/peekmyinterest/ProfileStorage.j
ava
@@ -95,6 +95,14 @@ abstract class ProfileStorage extends AppCompatActivity {
         }
     }

+    public void setNotes(String value) {
+        setTierOne("notes", value);
+    }
+
+    public String getNotes() {
+        return getTierOne("notes");
+    }
:
```

Undescriptive Commit Message



```
commit 53fc8b2a390e7b356e65f460a5d71917d5a3746c (origin/justin2, justin2)
Author: Jackson Leisure <jleisure@berkeley.edu>
Date:   Sat Dec 1 18:31:39 2018 -0800

    one edit boi

diff --git a/code/app/src/main/AndroidManifest.xml b/code/app/src/main/AndroidMa
nifest.xml
index eb97440..c75b07e 100644
--- a/code/app/src/main/AndroidManifest.xml
+++ b/code/app/src/main/AndroidManifest.xml
@@ -35,6 +35,7 @@
             android:label="@string/title_activity_personal_profile">

         </activity>
+        <activity android:name=".EditProfile" />
         <activity android:name=".MyProfile" />
         <activity android:name=".ReceivedProfile" />
         <activity android:name=".SentProfile" />
diff --git a/code/app/src/main/java/com/example/group39/peekmyinterest/Connect.j
ava b/code/app/src/main/java/com/example/group39/peekmyinterest/Connect.java
index 3fc9640..96db0b6 100644
--- a/code/app/src/main/java/com/example/group39/peekmyinterest/Connect.java
+++ b/code/app/src/main/java/com/example/group39/peekmyinterest/Connect.java
@@ -148,7 +148,7 @@ public class Connect extends ProfileStorage implements Outco
mingNfcManager.NfcAc



-        JSONObject fetchedJson = getMyJson(); //Fetches my json from profileSto
rage. (real code)
+        JSONObject fetchedJson = getJson(); //Fetches my json from profileStora
ge. (real code)
         //JSONObject fetchedJson = testJson;

         String fetchedJsonString = jsonToString(fetchedJson);
@@ -210,7 +210,7 @@ public class Connect extends ProfileStorage implements Outco
mingNfcManager.NfcAc

             this.tvOutcomingMessage.setText(inMessage); //Displays received Jso
n string in a textview.
:
```

Figure 2: Screenshots of example commits with "good" (upper) and "bad" (lower) messages associated with the changes in the commit. Note how the "good" commit message directly relates to the contents of the files which have been altered, while the "bad" message is vague and unrelated to the changes in the codebase.

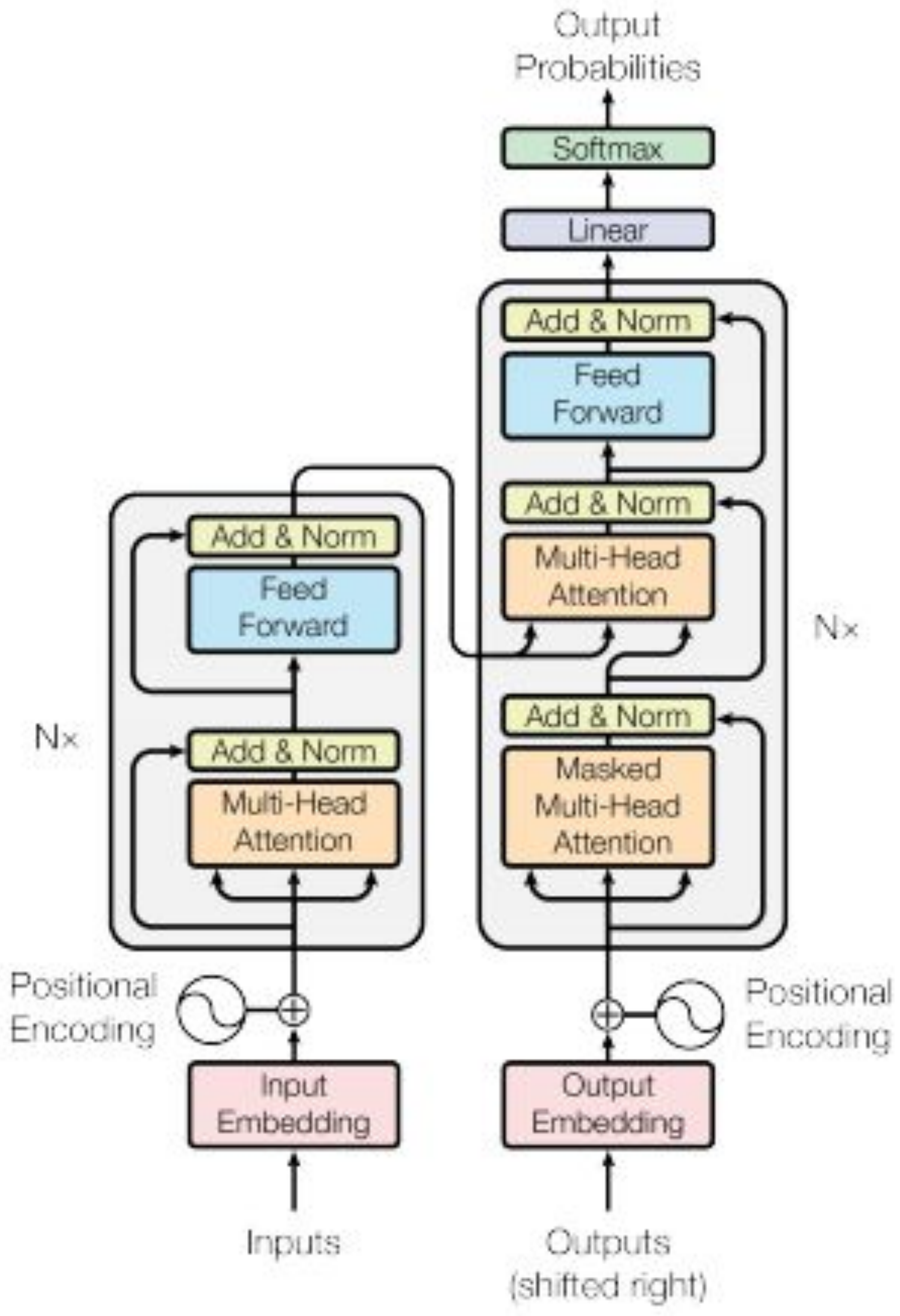


Figure 3: Diagram of the transformer architecture. We implement this architecture as a component of our summarization model (see 2nd model of Model Architecture).

Data Sources

- Wrote scripts to clone different public Java repositories and extract pairs of corresponding git diffs and commit messages
- Generally chose well-established repositories with high popularity
  - Training set: 15,000 diff-message pairs
  - Validation set: 3,600 diff-message pairs
  - Test set: 4,400 diff-message pairs
- Obtained git diffs and commit messages from 145 CS 61B student repositories from the Spring 2018 semester (used with permission; confidential)

Model Architecture

- Primarily experimenting with three models: an LSTM-based language model, a transformer-based summarization model, and state-of-the-art BERT
  - Language Model
    - Task: Input first n words of commit (ie. implemented, updated, added new feature, etc.) and generate rest of commit message
    - Potentially could help students autocomplete commit messages
    - LSTM cell architecture with cell size 256
  - Summarization Model
    - Task: Input git diff, generate and output a summary of the diff to represent a potential commit message
    - Transformer Architecture [1] with attentional RNN encoder-decoder (see Figure 3)
  - Bidirectional Encoder Representations from Transformers (BERT) Model [2]
    - Tasks
      - First, feed in diff-message pairs to determine if pair is true pair or negatively-sampled pair
      - Then modify/extend architecture to generate messages

Results

- Language Model Results
  - Example messages generated by model
    - "added a convenience method to the new api infrastructure"
    - "finished the test case for # version"
    - "update readme with release 2"
    - "changed the default value for the new api"
  - Loss converges to 4.5 after 15 epochs of training
- Summarization Model Results
  - Under development
- BERT Model Results
  - See Figure 4 below

```
{'auc': 0.5632171,
 'eval_accuracy': 0.8095839,
 'f1_score': 0.28571427,
 'false_negatives': 3514.0,
 'false_positives': 1465.0,
 'global_step': 8404,
 'loss': 0.641825,
 'precision': 0.3655262,
 'recall': 0.19366682,
 'true_negatives': 20325.0,
 'true_positives': 844.0}
```

Figure 4: Results outputted from BERT model on the discriminating pairs task (task 1). Further training is required, as misclassification is still fairly high.

References:

[1] Vaswani, Ashish et al. "Attention is All You Need" (Dec 2017): https://arxiv.org/pdf/1706.03762.pdf

[2] Devlin, Jacob et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" (Oct 2018): https://arxiv.org/pdf/1810.04805.pdf