Automatically Generating Commit Messages from Diffs using Neural Machine Translation

Siyuan Jiang, Ameer Armaly, and Collin McMillan
University of Notre Dame, USA
Commit Messages

<table>
<thead>
<tr>
<th>Author</th>
<th>Commit</th>
<th>Message</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yuya Nishihara</td>
<td>ba24165</td>
<td>setup: build Qt translation resource from precompiled files</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>3b87777</td>
<td>setup: compile Qt translation resource under build directory</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>3dce94</td>
<td>setup: do not pack unnecessary files to icon resource *.ico and *svg are no need</td>
<td>2015-06-07</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>72ec359</td>
<td>setup: wrap whole building process of Qt resource module by make_file()</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>b280b74</td>
<td>setup: build list of resource contents by caller We want to select files that will be used</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>578c750</td>
<td>setup: split stub function that builds icon resource</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>e203836</td>
<td>setup: inline compile_rc() to build_rc() Because future patch will wrap build_rc()</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>3d22b99</td>
<td>setup: look for pyroc4 relative to PyQt4 package on Windows On other</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>ca6bee9</td>
<td>setup: use make_file() and spawn() for mtime comparison and logging of</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>e28b22c</td>
<td>setup: clean up unused loop and defaults in build_qrc We don't have .qrc files in source</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>6998179</td>
<td>setup: simply propagate exception of uic.compileU() There should be no reason</td>
<td>2015-05-31</td>
</tr>
</tbody>
</table>
### Commit Messages

**GitHub**

#### Commits

<table>
<thead>
<tr>
<th>Author</th>
<th>Commit</th>
<th>Message</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yuya Nishihara</td>
<td>8e2416b</td>
<td>setup: build Qt translation resource from precompiled files</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>3b87477</td>
<td>setup: compile Qt translation resource under build directory</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>3dce9d4</td>
<td>setup: do not pack unnecessary files to icon resource <em>.ico and svg</em>.*svg are not used</td>
<td>2015-06-07</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>72ec359</td>
<td>setup: wrap whole building process of Qt resource module by make_file()</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>b280b74</td>
<td>setup: build list of resource contents by caller We want to select files that will be</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>578c750</td>
<td>setup: split stub function that builds icon resource</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>e203836</td>
<td>setup: inline compile_rc() to build_rc() Because future patch will wrap build_rc()</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>3d22b99</td>
<td>setup: look for pyrcc4 relative to PyQML package on Windows On other</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>ca6bee9</td>
<td>setup: use make_file() and spawn() for mtime comparison and logging of</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>e28b22c</td>
<td>setup: clean up unused loop and defaults in build_qrc We don't have .qrc files in fact</td>
<td>2015-05-31</td>
</tr>
<tr>
<td>Yuya Nishihara</td>
<td>6998179</td>
<td>setup: simply propagate exception of uic.compileUI() There should be no reason to</td>
<td>2015-05-31</td>
</tr>
</tbody>
</table>
Commit Messages

Many commit messages are similar[1][2]

Commit Messages

Many commit messages are similar[1][2]

Commit Messages

Many commit messages are similar\textsuperscript{[1]}\textsuperscript{[2]}

Commit Messages

Many commit messages are similar\cite{Mockus2000}\cite{Jiang2017}

\begin{itemize}
\item Remove unused images
\item Add test back to index
\item Update mock images
\end{itemize}

\begin{thebibliography}{99}
\end{thebibliography}
Commit Messages

Many commit messages are similar\textsuperscript{[1]}\textsuperscript{[2]}


Many commit messages are similar\cite{Mockus2000}\cite{Jiang2017}.

\begin{itemize}
\item GitHub
\item 2M commit messages
\item Update mock images
\item Neural Machine Translation (NMT)
\item Remove unused images
\item Add test back to index
\item Update mock images
\end{itemize}

\begin{itemize}
\end{itemize}
Neural Machine Translation (NMT)

Neural networks for translating natural languages, e.g. Chinese -> English

Neural Machine Translation (NMT)

Neural networks for translating natural languages, e.g. Chinese -> English

Neural Machine Translation (NMT)

Neural networks for translating natural languages, e.g. Chinese -> English

Parallel Corpus
- News articles
- Biomedical articles

Neural Machine Translation (NMT)

Neural networks for translating natural languages, e.g. Chinese -> English


Parallel Corpus

- News articles
- Biomedical articles
Neural Machine Translation (NMT)

Neural networks for translating natural languages, e.g. Chinese -> English

Parallel Corpus
- News articles
- Biomedical articles

Overview of Our Work

diffs ->
commit messages
Overview of Our Work

diffs ->
commit messages

Filter
Overview of Our Work

diffs -> commit messages

Filter

Neural Machine Translation (NMT)

Evaluation
Overview of Our Work

diffs ->
commit messages

Filter

Neural Machine Translation (NMT)

Evaluation

Quality Assurance Filter

Results
Overview of Our Work

diffs ->
commit messages

Filter

Neural Machine Translation (NMT)

Evaluation

Updated results

Quality Assurance Filter

Results
Preprocessing the Data Set

2M commit messages and diffs
- 1K most popular Java projects in Github

Preprocessing the Data Set

2M commit messages and diffs
- 1K most popular Java projects in Github*

Preprocessing the Data Set

2M commit messages and diffs
- 1K most popular Java projects in Github*

75K commit messages and diffs

Verb-Direct Object Filter

Verb-Direct Object is a phrase type

Verb-Direct Object Filter

Verb-Direct Object is a phrase type

Remove unused images

Verb-Direct Object Filter

Verb-Direct Object is a phrase type

Verb-Direct Object Filter

Verb-Direct Object is a phrase type

Verb-Direct Object Filter

Verb-Direct Object is a phrase type

47% of commit messages are begun with this type of phrases*

Verb-Direct Object Filter

Verb-Direct Object is a phrase type

47% of commit messages began with this type of phrases*

NLP Tool

- grammatical relations
- part-of-speech tags

32K commit messages and diffs

Verb-Direct Object Filter

Verb-Direct Object is a phrase type

47% of commit messages began with this type of phrases

NLP Tool
- grammatical relations
- part-of-speech tags

32K commit messages and diffs

Verb-Direct Object Filter

Verb-Direct Object is a phrase type

47% of commit messages began with this type of phrases

NLP Tool

- grammatical relations
- part-of-speech tags

NMT model: Nematus*

- Testing: 3K
- Validation: 3K
- Training: 26K

32K commit messages and diffs

Evaluation

Test Set

Trained NMT model

diff

References

Generated Commit Message

Commit Message

References

Generated Commit Message

Test Set
Evaluation

Test Set → Trained NMT model → diff → References → Commit Message → Similarity → Generated Commit Message
Evaluation

1. An automatic metric
2. A human study

Test Set

References

Commit Message

Similarity

Generated Commit Message

Generated Commit Message

Trained NMT model

diff
BLEU: the Automatic Metric

Bilingual Evaluation Understudy*
A popular metric for measuring the similarity between two sentences

BLEU: the Automatic Metric

Bilingual Evaluation Understudy*

A popular metric for measuring the similarity between two sentences

\[
\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} \frac{1}{N} \log(p_n)\right)
\]

BLEU: the Automatic Metric

**Bilingual Evaluation Understudy**

A popular metric for measuring the similarity between two sentences

\[
\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} \frac{1}{N} \log(p_n)\right)
\]

BLEU: the Automatic Metric

Bilingual Evaluation Understudy*
A popular metric for measuring the similarity between two sentences

\[ \text{BLEU} = \text{BP} \cdot \exp \left( \sum_{n=1}^{N} \frac{1}{N} \log(p_n) \right) \]

Brevity Penalty

**BLEU: the Automatic Metric**

Bilingual Evaluation Understudy*

A popular metric for measuring the similarity between two sentences

\[
\text{BLEU} = \text{BP} \cdot \exp\left( \sum_{n=1}^{N} \frac{1}{N} \log(p_n) \right)
\]

- Brevity Penalty
- Modified n-gram precision

4 (considers only 1 to 4-gram precisions)

---

BLEU: the Automatic Metric

* Bilingual Evaluation Understudy*

A popular metric for measuring the similarity between two sentences

$$\text{BLEU} = \text{BP} \cdot \exp\left( \sum_{n=1}^{N} \frac{1}{N} \log(p_n) \right) \in [0, 1]$$

Brevity Penalty

Modified n-gram precision

4 (considers only 1 to 4-gram precisions)

BLEU Results

Baseline: MOSES\textsuperscript{[1]}

Statistical machine translation system

# BLEU Results

**Baseline: MOSES**<sup>[1]</sup>

![MOSES](image)

<table>
<thead>
<tr>
<th>Model</th>
<th>BLEU (%)</th>
<th>$p_1$</th>
<th>$p_2$</th>
<th>$p_3$</th>
<th>$p_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOSES</td>
<td>3.63</td>
<td>8.3</td>
<td>3.6</td>
<td>2.7</td>
<td>2.1</td>
</tr>
<tr>
<td>NMT</td>
<td>31.92</td>
<td>38.1</td>
<td>31.1</td>
<td>29.5</td>
<td>29.7</td>
</tr>
</tbody>
</table>

### BLEU Results

**Baseline: MOSES[^1]**

<table>
<thead>
<tr>
<th>Model</th>
<th>BLEU (%)</th>
<th>$p_1$</th>
<th>$p_2$</th>
<th>$p_3$</th>
<th>$p_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOSES</td>
<td>3.63</td>
<td>8.3</td>
<td>3.6</td>
<td>2.7</td>
<td>2.1</td>
</tr>
<tr>
<td>NMT</td>
<td>31.92</td>
<td>38.1</td>
<td>31.1</td>
<td>29.5</td>
<td>29.7</td>
</tr>
</tbody>
</table>

Most Diffs: ≥ 75 words
Most Messages: < 30 words

Human Study

BLEU

• Two sets of sentences
• Textual similarity
Human Study

BLEU
• Two sets of sentences
• Textual similarity

Human Study
• Individual sentences
• Semantic similarity
Human Study

Survey
20 Programmers
Human Study

Below are two commit messages,

*Message 1:* Updated changelog
*Message 2:* Fix snapshot version

How **similar** are the two messages (in terms of the **meaning**)?

0 1 2 3 4 5 6 7
no similarity whatsoever identical

(Optional) **Justification:**
Human Study

983 pairs of generated/reference messages were rated:
• 226 pairs by three programmers
• 522 pairs by two programmers
• 235 pairs by one programmer
Human Study

(semantic similarity: 0-no similarity, 7-identical)
Human Study

(semantic similarity: 0-no similarity, 7-identical)
Human Study

(semantic similarity: 0-no similarity, 7-identical)
Human Study

(semantic similarity: 0-no similarity, 7-identical)
Quality Assurance Filter

Data: 983 commits that were evaluated in the human study
Quality Assurance Filter

Data: 983 commits that were evaluated in the human study
Quality Assurance Filter
Quality Assurance Filter

Detected 44% of the bad cases
Summary

diffs -> commit messages

Filter

Neural Machine Translation (NMT)

Evaluation

Updated results

Quality Assurance Filter

Results
Summary

diffs -> commit messages

Neural Machine Translation (NMT)

Generate short commit messages that are high-level overviews of software changes

Updated results

Quality Assurance Filter

Evaluation

Updated results

Results
On the Job Market

- **Software Engineering**, Program Comprehension
- Data Science
- Machine learning

✉️ sjiang1@nd.edu