Unsupervised Machine Translation and Repair

Colt Dahl
dahlcolt@gmail.com
Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA

Abstract

Effective communication across languages is gaining importance as the the worldwide community interacts more and more frequently within an interlingual context. It is time consuming to train individuals to learn a language, effort consuming to produce translations in the traditional sense through two-way translators, and expensive due to these foreign language skills being in high demand. Using new methods in natural language processing, we can produce translations using less corpora, fewer dedicated translators, and an equivalent or lessened measure of time. This paper explores these methods. This paper starts by introducing more general approaches, such as supervised machine translation, and then focuses on less known novel approaches. First, it introduces the baseline methods of statistical machine translation and neural machine translation, then it describes more advanced tools like AdvGen and TransRepair.

1 Introduction

Machine translation (MT) is the use of software to interpret text from one language into another without the use of any human translator. Before the advances of machine learning, the field for research into machine translation has been Rule-based. Rule-based machine translation (RBMT) uses relationships between languages explicitly written into the translator by technicians or linguists and requires heavy supervision and development time, not counting any lost subtleties within any language that could not be reasonably taken into account using a series of rules. For example, sarcasm within text is a highly complex lexical rule to attempt to construct manually.

1.1 Statistical Machine Translation

SMT, or statistical machine translation, relies entirely on bilingual corpora. Bilingual corpora are a language resource containing documents that are translations of each other. These types of data are frequently known as parallel texts, as they seek to match phrases between languages. The method attempts to match source language phrases with the most similar phrases in the target language, finding the probability that a target string is a translation of the source string, and the probability of seeing that string within the same context within the corpora. As the string appears more and more in the corpora in that context, it is more and more likely to be predicted to finish a text. The studies referenced in this paper explore making use of generated synthetic parallel data for use within machine translation. [4]

1.2 Neural Machine Translation

NMT, or neural machine translation, is a predictive machine translation system using neural networks. Neural networks take input data, in our case, what we wish to translate, and that input goes through a series of “hidden“ nodes which act as the main computational method. The output from these hidden nodes is then compared to the expected output from the neural network’s input. If the output is not a good match to the expected output (known as training data), the hidden nodes change their computational method so as to make that output less likely with that input, and the opposite happens if the output is a good match. Neural machine translation attempts to predict the probability of a sequence of words using this training data and it typically translates more effectively than statistical machine translation. Recurrent neural networks are neural networks which form a cycle, feeding previous neural network outputs into neural inputs to assist in training, typically increasing the quality of translation.

1.3 Unsupervised Machine Translation

UMT, or unsupervised machine translation, is a method that uses fewer manually produced parallel text pairs as parallel data for training than supervised MT. All methods previously described use supervised machine translation. For our purposes, “unsupervised" defines the training of MT systems on parallel texts where one side is synthetic. Starting with a small development set of manually produced sentence pairs, we generate synthetic parallel data from monolingual data and this is used by the machine translation method.

1.4 BLEU

BiLingual Evaluation Understudy (BLEU) is an algorithm for determining closeness to a human translator [5] . It is an effective benchmark for determining if a method of machine translation can successfully match or come close to matching human judgement. The higher the number, the higher the closeness to a perfect translation. 0 meaning completely imprecise and 100 meaning completely precise. The input is what the machine translator takes in, the output is the machine translation output, and the references are two good quality translations to be used as comparison to the output.

Example 1

- Input: “Le chat est sur le tapis.“
- Reference 1: “The cat is on the mat.“
- Reference 2: “There is a cat on the mat.“
- Output: “The cat the cat on the mat.“

Starting with example 1, an n-gram is a sequence of n words, a unigram is a sequence of 1 word, a bigram is a sequence of 2 words, and so on, we isolate unigrams in the candidate sentence and compare them to the good quality reference sentences. The algorithm checks each word in the candidate phrase (labeled as Output) to see if it appears in the references, and if so, it gets a point. As each word in the candidate sentence can be found in the references, we ignore multiples and only count towards the score if it hasn’t already appeared. We can aim for a more accurate measure of precision by changing the unigrams to bigrams. Now the output sentence is isolated into sequences of 2 words, i.e. “the cat“, “cat the“, and so on. "the cat" appears in both references, “cat the“ appears in none, “cat on“ appears in 1, “on the“ appears in both, and “the mat“ appears in both. The total number of bigrams in the candidate sentence is 6 and the total unique bigrams in the candidate sentence matching references is 4, for 4/6: for a BLEU score of 66.67. BLEU goes further than bigrams and unigrams, typically using sequences of 4 words for precision calculation. The BLEU precision scores for sequences are combined using geometric means to prevent shorter translations from receiving high precision scores too easily.

2 Methods

2.1 Iterative training of unsupervised SMT

Proposed by the referenced paper, an alternative framework to improve unsupervised SMT (USMT) and unsupervised NMT (UNMT) systems iteratively. Assuming that the UNMT approach can produce translations of a better quality than those by USMT, a new USMT system performs using synthetic parallel data generated by the previous UNMT system and is expected to create a system better than the previous USMT system. This method can be repeated for several iterations to improve USMT and UNMT. Within each iteration, training is done without the use of previously generated synthetic data, as to lessen possible bias. The original training of UNMT is initialized by the synthetic parallel data generated by the previous USMT system, and vice versa, except for the very first USMT initialization, which uses a synthetically produced data from a set of good quality manually produced phrases.

Phrase table induction [4], here used as a source of natural language data, is a resource that takes a set of manually produced phrases, extracts sequences of a length up to seven words from the monolingual corpora of manually produced good translations, and induces the phrase tables by attempting to extract matching or similar phrases from the manually produced set. This method provides many source phrases for us to work with. These phrases that are extracted are meaningful due to the way they are collected: the system attempts to identify a sequence of words as a phrase by the frequency the words appear together, the higher the frequency, the more likely a phrase will be identified as such. [4]

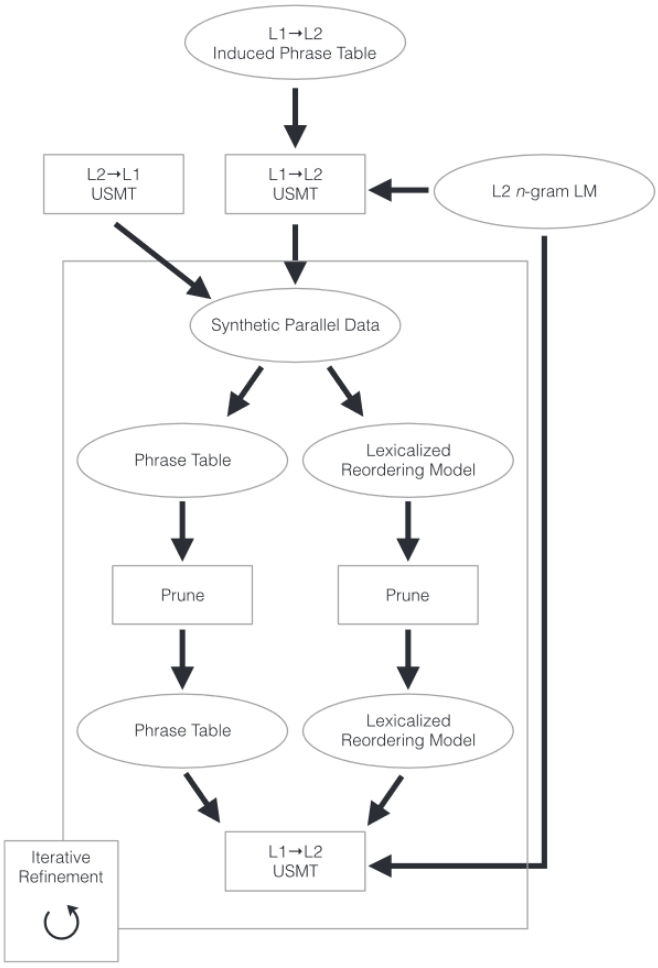

Figure 1. USMT Framework for a given language pair L1-L2.[4]

Figure 1 shows the diagram of USMT in this framework.

Induced Phrase Table: Phrase pairs are calculated from the similarity scores of the pairs, calculated from local contextual similarity [7]. Afterwards, automated refinement of the unsupervised SMT begins, as otherwise, translation between very distant languages would be very poor.

Lexical Reordering Model: The reordering model is learned from the parallel text data through the relations of adjacent words in sequence. For example, English is a subject-verb-object language. “John eats pie“ is a valid sentence. John, the subject, eating, the verb for which our subject is acting on, and the object, what our subject is acting on itself. If English were a subject-object-verb language, “John eats

pie“ would be transformed into “John pie eats“ after being lexically reordered. The optimal alignment for the given word order and the word-to-word translation possibilities are computed through the learned model and once this reordering/realignment is finished, the resulting aligned sentence pairs are more accurately correlated to each other than before.

Pruning: The synthetic data is pruned as proposed by previous work through the use of significance testing. Significance testing assesses associations by calculating the probability that an observed table could occur by chance, for example, “I read a book.“ would appear more frequently than “I read an apple.“, both the latter and former would appear randomly, but because “I read a book.“ is so much more common than “I read an apple.“, despite them both appearing randomly, we are able to determine that the first phrase is significant because of how unlikely it would be that it could occur that many times naturally, i.e. it’s the difference between seeing one perfectly spherical rock and seeing one thousand perfectly spherical rocks on your beach visit. Up to 90% [3] of phrase pairs are pruned due to low-quality or non-use in any translation, without any significant reduction to BLEU score.

The pruned phrase table is then plugged into the USMT system, and after translation, iterated into an unsupervised neural machine translation system (UNMT).

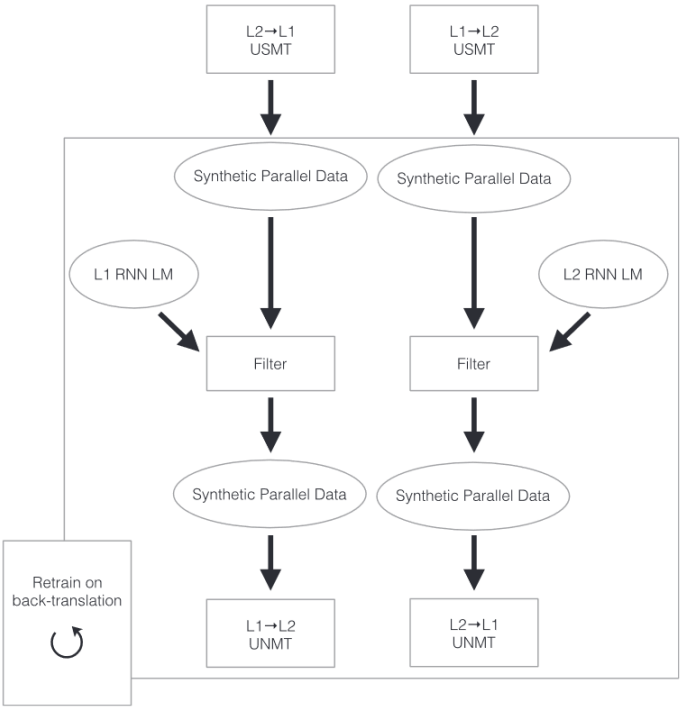

Figure 2. UNMT Framework for a given language pair L2-L1 [4].

2.2 Iterative training of unsupervised NMT

The synthetic parallel data taken from the previous USMT system is now being used as the baseline parallel data for the UNMT method. It has the same goal as the previous USMT, to maximize the likelihood of parallel phrase detection and filtering. To obtain better translations, UNMT integrates the synthetic parallel data of USMT as mentioned before.

Figure 2 shows the diagram of UNMT in this framework.

Synthetic Parallel Data: The starting parallel data is constructed using the previously produced synthetic parallel data from USMT.

Filtering: Since USMT commonly outputs ungrammatical translations, only the sentence pairs with the highest fluency are kept. Deciding which pairs are kept is the normalized language model score, a monolingual model. Using the synthetic parallel data from the earlier USMT iterations, a monolingual dataset, and a recurrent neural network to filter the least grammatical 33% of pairs produced to keep even fewer sentence pairs for training.

The filtered parallel data is then plugged into the UNMT system, training the L1→L2 UNMT system on synthetic parallel data generated by back-translating L1 sentences using the L2→L1 UNMT system, as well as those generated by the L2→L1 USMT system. Back-translation is: transforming the translated document back into its original language. Translation quality improves and the cycle recurs, using the UNMT parallel data generated and filtered back into the USMT system.

2.3 Automated repair of self-learning

Even with so many positive considerations and outcomes from NMT, perturbations within input sentences can force translation output to become entirely erroneous only with small changes and errors. Natural noise and other errors must therefore be dealt with in a way that allows the output to be understood even with these errors. Neural networks are trained to learn using a combination of erroneous and correct material.

TransRepair is a tool meant to test and repair MT in a novel way [8]. It generates tests for checking whether an inconsistency bug has been found after taking the original sentence/translation and mutating the original sentence via word replacement. This process is done multiple times, consistency score is calculated by how often the words are used interchangeably, automatically filtering mutants by score. For example, BLEU scores, mentioned previously, are a consistency metric, another metric is the LCS-based metric, which measures the similarity of two sequences by their longest common subsequence, given that they appear in the same relative order. The original translation is then compared to the mutated translations in one of the previously mentioned consistency metrics and if an inconsistency is discovered through a low consistency score in a consistency

metric mentioned above, it automatically repairs the inconsistent translation in line with the other mutants to raise consistency score. TransRepair uses the earlier mutations and methods of text prediction or cross references in an attempt to fix the original translation and output an optimal translation. The text prediction probability does not require the training data or any code from the original translation method's training algorithm, just the provided tool. The cross-reference method only requires the execution of the translator and its output, making TransRepair an adaptive way of repairing translations, due to it not needing the training data or any code from the training algorithm.

2.4 AdvGen

An ideal NMT model would generate similar translations for inputs that are similar to each other, but this is not always the case. A translation may be incorrectly different through small changes in the input sentence, changing the meaning of the sentence entirely. AdvGen is a supervised neural machine translation tool, to be used with neural machine translators, meant to improve this model translation "robustness" [2]. Increasing robustness, how perturbed a pair may be and still output a valid translation, is our goal. To do this, AdvGen generates plausible examples of erroneous translations by randomly selecting some words in a sentence, checking those words and a list of closely associated words. Whichever word in the sentence is most erroneous based on how unlikely it is to appear in that context, replaces the word in the sentence while feeding it back to the neural network model for testing. These are expected to retain some level of similarity but still be different enough that it may confuse the system. This new sentence is now a new data point to improve the neural model's robustness.

3 Results

3.1 Iterative training of USMT/UNMT

Figure 3 shows the results of the iterative training method for USMT/UNMT. The languages are de: German, en: English, fr: French, and ja: Japanese. Newstest and NTCIR are language data sets, Newstest is news data and NTCIR is data from the National Institute of Information Test Collection for Information Resources. The number after the model type is the iteration number.

The method of iterative training can bring as much improvement as adding twice as many sentence pairs for training. BLEU scores of 15 to 25 (in translations between languages of similar etymologies) were obtained through methods used in prior research, such as the Lample method, the predecessor to the current method. Both use a hybridized system of statistical and neural machine translation to produce parallel data. The iterative training method of USMT was able to produce BLEU scores in the 20 to 28 range, a marked improvement.

System	Newstest				NTCIR		
	de→en	en→de	fr→en	en→fr	ja→en	en→ja	#
Lample et al. [24]'s USMT	22.1	17.5	26.2	23.9	20.5	21.6	1
Lample et al. [24]'s UNMT	20.3	17.0	23.6	22.9	15.8	17.2	2
USMT-1	23.4	18.8	26.7	25.3	21.3	22.0	3
↳ UNMT-1	29.4	22.8	28.8	28.1	25.3	27.8	4
↳ USMT-2	26.6	21.4	28.0	27.3	21.6	25.0	5
↳ UNMT-2	**30.4**	**24.3**	**29.2**	**29.0**	25.9	**29.2**	6
UNMT-1 ($P = 8.5 \times 10^6$)	29.8	22.8	28.9	28.4	**26.0**	28.0	7
Supervised SMT	30.4	26.4	35.3	32.7	27.6	31.3	8
Supervised NMT	35.8	32.9	35.9	37.2	43.5	48.7	9

Figure 3. Comparison of BLEU iterative hybrid system and previous Lample framework scores for the unsupervised iterative method, along with comparisons to supervised SMT and NMT scores, Newstest and NTCIR are language data sets. [4]

The iterative system outperformed the Lample method for all tasks, even going as far as the iterative USMT performing better than all Lample UNMT tasks. Consistent improvement over the iterations can be seen, with UNMT-2 and USMT-2 both showing more effective translation than their UNMT-1 and USMT-1 counterparts, and for some tasks, the unsupervised machine translations were able to gain a result nearly as good as the supervised machine translations.

The difference is especially pronounced when comparing the Lample et al. method of UNMT to the novel method of UNMT, we can see increases of 10 or more BLEU score in the en⟶ja (English to Japanese) and ja⟶en result columns.

Time wise, it took 4 days for the first iteration's training to complete, and then 6 days for the second iteration, while Lample et al. varied from 2 to 12 days. Training time appears to be not significantly differ from the previous method.

3.2 TransRepair

Figure 4: translation improvement with TransRepair. Translation acceptability as scored by manual inspection, the first four rows are using the translation model of Google Translate, and the second four rows are TransRepair using a similarity metric based on longest common sequence, and the third series of rows are the probability-based approach. We can see that comparatively, TransRepair shows consistent translation improvement from manual inspection, and it is shown to have a lower translation repair cost because it does not require model retraining. Compared with training approaches, TransRepair shows consistent translation improvement from manual inspection.

3.3 AdvGen

Figure 5 shows the comparison of baseline NMT BLEU scores by model to AdvGen. Experiments were conducted with the LDC (Linguistic Data Consortium) corpus of 1.2M sentence pairs for Chinese-English, and the 2014 WMT (Workshop on Machine Translation) corpus of 4.5M English-German

	Aspect	Improved	Unchanged	Decreased
GT.LCS	**Translation consistency**	33 (85%)	4 (10%)	2 (5%)
	Translation acceptability: overall	22 (28%)	48 (62%)	8 (10%)
	Translation acceptability: original	10 (26%)	23 (59%)	6 (15%)
	Translation acceptability: mutant	12 (31%)	25 (64%)	2 (5%)
Trans.LCS	**Translation consistency**	24 (89%)	3 (11%)	0 (0%)
	Translation acceptability: overall	15 (28%)	37 (69%)	2 (4%)
	Translation acceptability: original	7 (26%)	19 (70%)	1 (4%)
	Translation acceptability: mutant	8 (30%)	18 (67%)	1 (4%)
Trans.Prob	**Translation consistency**	51 (88%)	6 (10%)	1 (2%)
	Translation acceptability: overall	30 (26%)	76 (66%)	10 (9%)
	Translation acceptability: original	15 (26%)	36 (62%)	7 (12%)
	Translation acceptability: mutant	15 (26%)	40 (69%)	3 (5%)

Figure 4. Translation improvement with TransRepair [8]

Method	Model	MT06	MT02	MT03	MT04	MT05	MT08
Vaswani et al. (2017)	Trans.-Base	44.59	44.82	43.68	45.60	44.57	35.07
Miyato et al. (2017)	Trans.-Base	45.11	45.95	44.68	45.99	45.32	35.84
Sennrich et al. (2016a)	Trans.-Base	44.96	46.03	44.81	46.01	45.69	35.32
Wang et al. (2018)	Trans.-Base	45.47	46.31	45.30	46.45	45.62	35.66
Cheng et al. (2018)	RNMT$_{lex.}$	43.57	44.82	42.95	45.05	43.45	34.85
	RNMT$_{feat.}$	44.44	46.10	44.07	45.61	44.06	34.94
Cheng et al. (2018)	Trans.-Base$_{feat.}$	45.37	46.16	44.41	46.32	45.30	35.85
	Trans.-Base$_{lex.}$	45.78	45.96	45.51	46.49	45.73	36.08
Sennrich et al. (2016b)*	Trans.-Base	46.39	47.31	47.10	47.81	45.69	36.43
Ours	Trans.-Base	46.95	47.06	46.48	47.39	46.58	37.38
Ours + BackTranslation*	Trans.-Base	**47.74**	**48.13**	**47.83**	**49.13**	**49.04**	**38.61**

Figure 5. Comparison of NMT BLEU scores by model to AdvGen [2]

sentence pairs. The NIST (National Institute of Standards and Technology) 2002, 2003, 2004, 2005, and 2008 sets were used as test sets: sets used to assess the final model's quality separate from the training data. The robustness of NMT models improves using the AdvGen tool, results from perturbed Chinese→English and English→Chinese translations show that you can output similar results to unperturbed translations with AdvGen, improving BLEU scores. The AdvGen approach, here listed as Ours+Back-Translation, achieves an average gain of 2.25 BLEU score and up to 2.8 BLEU score maximum. A back-translated corpus is incorporated to increase the model's precision, as discussed in Section 3.2. Since all methods were built on top of the same backbone, "Trans.-Base", or baseline Transformer, this shows the efficacy of AdvGen.

4 Conclusion

The study of unsupervised MT and methods for selecting and inducing synthetic parallel data are bolstered by the methods referenced in the section on the iterative training of unsupervised machine translators. Using a small amount of manually produced parallel phrases to improve unsupervised translation and synthetic parallel data continues to be worked on by the researchers [1].

TransRepair automatically tests and improves context-similar translation, while previous work relied largely on adding noisy data to the training set and then retraining the model, AdvGen is an example of this.

Mentioned as the next step by the research team behind AdvGen in their conclusion, curriculum learning [6] uses a system for estimating difficulty of a sample sentence for translation, and shows the model these samples at certain times based on their difficulty.

In this paper, we presented an iterative approach for training of statistical and neural machine translation, AdvGen, and TransRepair. All three showed improvements in translation quality and improved closeness to human translation.

Acknowledgments

The feedback given by Elena Machkasova, of the Science and Mathematics division at the University of Minnesota, Morris, as well as the feedback given by Sydney Richards, an alumni of the Science and Mathematics division at the University of Minnesota, Morris, was invaluable in the creation of this paper.

References

[1] [n.d.]. Benjamin Marie's website. http://benjaminmarie.com/. Accessed: 2021-04-15.

[2] Yong Cheng, Lu Jiang, and Wolfgang Macherey. 2019. Robust Neural Machine Translation with Doubly Adversarial Inputs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 4324–4333. https://doi.org/10.18653/v1/P19-1425

[3] Howard Johnson, Joel Martin, George Foster, and Roland Kuhn. 2007. Improving Translation Quality by Discarding Most of the Phrasetable. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. Association for Computational Linguistics, Prague, Czech Republic, 967–975. https://www.aclweb.org/anthology/D07-1103

[4] Benjamin Marie and Atsushi Fujita. 2018. Phrase Table Induction Using Monolingual Data for Low-Resource Statistical Machine Translation. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* 17, 3, Article 16 (Feb. 2018), 25 pages. https://doi.org/10.1145/3168054

[5] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA, 311–318. https://doi.org/10.3115/1073083.1073135

[6] Emmanouil Antonios Platanios, Otilia Stretcu, Graham Neubig, Barnabas Poczos, and Tom Mitchell. 2019. Competence-based Curriculum Learning for Neural Machine Translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, Minnesota, 1162–1172. https://doi.org/10.18653/v1/N19-1119

[7] Yangqiu Song and Dan Roth. 2015. Unsupervised Sparse Vector Densification for Short Text Similarity. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Denver, Colorado, 1275–1280. https://doi.org/10.3115/v1/N15-1138

[8] Zeyu Sun, Jie M. Zhang, Mark Harman, Mike Papadakis, and Lu Zhang. 2020. Automatic Testing and Improvement of Machine Translation. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering* (Seoul, South Korea) *(ICSE '20)*. Association for Computing

Machinery, New York, NY, USA, 974–985. https://doi.org/10.1145/ 3377811.3380420