Tune In: The AFRL WMT21 News-Translation Systems

Grant Erdmann, Jeremy Gwinnup, Timothy Anderson
Air Force Research Laboratory
{grant.erdmann, jeremy.gwinnup.1, timothy.anderson.20}@us.af.mil

Abstract

This paper describes the Air Force Research Laboratory (AFRL) machine translation systems and the improvements that were developed during the WMT21 evaluation campaign. This year, we explore various methods of adapting our baseline models from WMT20 and again measure improvements in performance on the Russian–English language pair.

1 Introduction

As part of the 2021 Conference on Machine Translation (wmt, 2021) news-translation shared task, the AFRL human language technology team participated in the Russian–English portion of the competition. We experiment with OpenNMT-tf [1] (Klein et al., 2018) and Marian [2] (Junczys-Dowmunt et al., 2018) transformer (Vaswani et al., 2017) models trained as part of our WMT20 (Gwinnup and Anderson, 2020) efforts and apply varying continued-training and fine-tuning approaches (Luong and Manning, 2015; Freitag and Al-Onaizan, 2016), including a new method to select a fine-tuning set from a separate, larger corpus not used in training.

We submit an OpenNMT-based transformer system fine-tuned on newstest test sets from 2014-2017 as our primary entry, and a Marian-based transformer system fine-tuned on newstest test sets from 2014-2018 as a contrast.

2 Data and Preprocessing

Since most of our efforts focus on fine-tuning existing models this year, we reuse the training corpus from our WMT20 systems which includes the following parallel corpora: Commoncrawl (Smith et al., 2013), Yandex[3], UN v1.0 (Ziemski et al., 2016), Paracrawl[4](Esplà et al., 2019), Wikimatrix (Schwenk et al., 2019), and backtranslated data from our WMT17 system (Gwinnup et al., 2017) as well as Edinburgh's WMT17 system (Sennrich et al., 2017) yielding a raw corpus of over 76.3 million lines.

The new Russian–English version 8 Paracrawl corpus is reserved for tuning set selection as described in Section 2.3.

2.1 Data Preparation

We re-use the fastText (Joulin et al., 2016b,a) based language ID filtered corpus with an ID threshold of 0.8 as described in Gwinnup and Anderson (2020), shown in Table 1, allowing us to make concrete progress comparisons to last year's systems.

2.2 Data Augmentation with Speech Recognition-like output

In order to build a larger pool of training data, we have created Automatic Speech Recognition (ASR) - like training data for the Russian–English translation task. Whereas written text can include upper and lowercase characters, punctuation, special symbols, and numbers written using digits, transcripts produced by ASR systems are typically uncased with no punctuation, no special symbols, and numbers written as spoken (e.g., 4.1% rendered as "four point one percent"). In previous experiments on an English-German spoken language translation task (Ore et al., 2020), we found that we could get an improvement in BLEU score by formatting the MT training data such that the source language text matched the output format of our ASR system, while leaving the target language text unmodified. We applied a similar procedure to the Russian side of the Russian-English training corpus using the text2norm.pl script from ru2sphinx.[5] This copy of the ASR-like training text was then appended to

[1] Available at: https://github.com/OpenNMT/OpenNMT-tf/

[2] Available at: https://github.com/marian-nmt/marian

[3] https://translate.yandex.ru/corpus?lang=en

[4] Version 1 Russian–English parallel data

[5] Available at: https://github.com/zamiron/ru4sphinx

Proceedings of the Sixth Conference on Machine Translation (WMT), pages 110–116
November 10–11, 2021. ©2021 Association for Computational Linguistics

corpus	unfiltered lines	filtered lines	percent remain
commoncrawl	723,256	655,069	90.57%
news-commentary-v15	319,242	286,947	89.88%
yandex	1,000,000	901,318	90.13%
un-2016	11,365,709	9,871,406	86.85%
paracrawl-v1	12,061,155	5,173,675	42.90%
wikimatrix	5,203,872	4,287,881	82.40%
wmt17-afrl-bt	8,921,942	8,317,107	93.22%
wmt17-uedin-bt	36,772,770	29,074,022	79.06%
Total	76,367,946	58,567,425	76.69%

Table 1: Results of language-id based Russian–English corpus filtering with threshold of 0.8 as reported in (Gwinnup and Anderson, 2020)

the original training data, effectively doubling the size of the corpus.

2.3 Selecting Tuning Sets from Representative Data

We performed experiments involving automatic selection of fine-tuning corpora. Given a monolingual application corpus, we wish to test the possibility of selecting an appropriate fine-tuning set to improve a general-purpose neural MT system's performance on that application corpus. We anticipate such techniques to be of increasing importance, especially for high-value application corpora, as computational costs of subcorpus selection and fine-tuning continue to decrease.

2.3.1 Method

We performed subselection as in Erdmann and Gwinnup (2019), which can flexibly incorporate a text quality metric and multiple parallel text corpora. In short, this algorithm tries to simultaneously optimize the quality of the subset's text and its coverage of the vocabulary present in given application corpora.

Of special note is our use of clustering to select data. We hierarchically applied the MAPPER algorithm (Singh et al., 2007) to cluster sentence vectors of a monolingual corpus. The clusters deemed useful were then used to assign fuzzy clustering to the application corpus and the corpus from which we subselect. This clustering information was included as one of the text corpora.

2.3.2 Application

The application corpus we used was the Russian side of newstest2019 and newstest2020, totalling 6777 lines. The pool of possible parallel text for subselection we took to be the given 12.6M-line subset of Russian–English version 8 ParaCrawl corpus with LASER score at least 1.1. For subselection algorithms, we first preprocessed the Russian text, applying a 90k-element joint BPE. We used the algorithm in Erdmann and Gwinnup (2019) to subselect a corpus, using 3-grams in the vocabulary coverage. As a text quality metric in this algorithm we used either the provided Bicleaner scores (Sánchez-Cartagena et al., 2018; Ramírez-Sánchez et al., 2020) or the word-averaged scores provided by OpenNMT's scoring functionality, using the untuned OpenNMT model we developed for this year's task. In order to provide meaningful comparisons with our baseline fine-tuning set of newstest2014-2018, we matched its size by always subselecting a fine-tuning set with fifteen thousand lines. Fine-tuning was performed using a single-model Marian-based untuned MT system as a baseline.

Sentence vector clustering was learned using a 570M-line monolingual Russian corpus built from the concatenation of monolingual CommonCrawl (Smith et al., 2013) data provided by WMT organizers as part of our WMT18 efforts towards pretraining word embeddings. The word vectors were trained using word2vec (Mikolov et al., 2013) on this corpus, after applying a 90k-element joint BPE. These embeddings have a dimensionality of 512 to match our Marian transformer-base system configuration as described in Gwinnup et al. (2018). A randomly-chosen 100k-line subset of the corpus was used to find the clustering.

Several methods of converting word vectors to sentence vectors were considered, and we empirically chose a "softened sum" of the word vectors

w_i as the sentence vector s:

$$s = \frac{\sum w_i}{\log(1 + \text{number of words in sentence})}.$$

Clusters were considered to be useful if they covered between 1% and 5% of this corpus. In this case there were 19 such clusters, having between 1000 and 5000 representatives each. These clusters were found to have qualitative meaning to a Russian linguist: clusters with relatively high representation in our application corpus tended to be news-like, and clusters with relatively high representation in ParaCrawl tended to be noisier.

We computed membership of a given sentence vector in a fuzzy clustering sense, with weight of cluster i defined as

$$z_i = (\min \text{distance}/\text{distance}_i)^4$$

where we use Euclidean distance, and the minimum is taken over all 19 clusters. Although the exact form is empirical, note that the weight has a maximum of unity at the closest cluster and that a cluster will get lower weight if it is farther from the sentence vector. This fuzzy clustering was computed once using k-means (distance is to cluster mean) and once using single-linkage (distance is to nearest member) clustering. These two membership clusters were then averaged. Coverage of the clusters was encouraged by including the clustering as another text corpus in our standard algorithm (Erdmann and Gwinnup, 2019) — each sentence vector was converted into a 100-word "sentence," where each cluster's "word" appeared a number of times relative to the magnitude of its weight in the line's clustering[6]. Naturally, coverage of these clustering words was computed using only unigrams.

2.3.3 Results

Table 2 shows the results of our fine-tuning experiments. The "clustering" and "metric" columns designate whether clustering was incorporated and whether Bicleaner ("Bic") or NMT scoring ("NMT") was used as the text quality metric. We see consistent gains over the untuned set, even on newstest2021, which was not used in the selection. The three subselection methods produced similar results on the three test sets. Fine-tuning with our selected sets did not produce consistent improvement over our baseline fine-tuning using newstest2014-2018. Compared to this baseline fine-tuning, the new sets improved performance on newstest2019 (roughly $+0.7$ BLEU), but they lowered performance on newstest2020 (roughly -0.7 BLEU) and the unseen newstest2021 (roughly -1.1 BLEU). Our generated fine-tuning sets did not show a consistent benefit for this task, so they were not used in our submission systems. Without further information, we cannot attribute the quality of the results to the method, the quality of data in ParaCrawl, or other causes.

Our method generates a pseudo in-domain set for an unknown application domain, using only source-side data of the application corpus. This generated set can be used for fine-tuning, training, or other purposes in natural language processing. We believe that such techniques warrant further investigation, especially for an application corpus where the domain is unknown or human-curated parallel data are unavailable.

3 Machine Translation Systems

3.1 OpenNMT-tf

The OpenNMT-tf system trained for this task used the configuration for a big deep transformer network.

We used the following network hyperparameters:

- 1024 embedding size
- 4096 hidden units
- 12 layer encoder
- 12 layer decoder
- 16 transformer heads
- dropout 0.3
- attention dropout 0.1
- feed forward network dropout 0.1
- embeddings for source, target and output layers were not tied
- Layer normalization
- Label smoothing 0.1
- Learning rate warm-up 8000 steps

The corpus used for the initial model consisted of commoncrawl, paracrawl v1, and news-commentary-v13 from wmt19 and was processed

[6] For example, using a 10-word sentence for brevity, this process would convert the fuzzy cluster membership vector $[0.2, 0.0, 0.8, 1.0]$ into the sentence "0 2 2 2 2 3 3 3 3 3".

tuning set	clustering	metric	newstest2019	newstest2020	newstest2021
untuned			35.9	34.5	46.5
newstest2014-2018			37.5	35.7	49.3
selected	no	NMT	38.0	35.0	48.4
selected	no	Bic	38.3	35.0	48.2
selected	yes	Bic	38.2	34.9	47.9

Table 2: Tuning sets and resultant BLEU scores.

with SentencePiece(Kudo and Richardson, 2018) using a model with a vocabulary size of 40K trained on this ru-en corpus of 16,805,109 bi-text. This was one of our WMT20 submitted systems (Systems 3 and 4 in Table 3). Additionally the corpus was processed as described in Section 2.2 to resemble ASR output and the resulting data was combined with the above for a final count of 33,610,218 bi-text. The network was trained for 10 epochs of this training data using a batch size of 3124 and an effective batch size of 49984 using the lazy Adam (Kingma and Ba, 2015) optimizer with beta1=0.9, beta2=0.998 and learning rate 2.0. This was a system that had been originally trained for speech translation application but showed improvements in text translation as well. The final submitted system continued training an additional 2 epochs using the unfiltered data described in Table 1. This was done to try to take advantage of the larger data set and not having the computational resources or time to train a new system with with the larger data set in time for submission deadline. The output was an average of the last 8 checkpoints of training. Checkpoints were saved every 5000 steps. The system was then tuned with three epochs of newstest data from years 2014-2017 (Systems 5 and 6 in Table 3).

3.2 Marian

Our Marian systems utilize the transformer architecture in the transformer-base configuration. We use the WMT14 newstest2014 test set for validation during training and the following network hyperparameters:

- 512 embedding size
- 2048 hidden units
- 6 layer encoder
- 6 layer decoder
- 8 transformer heads
- Tied embeddings for source, target and output layers
- Layer normalization
- Label smoothing
- Learning rate warm-up and cool-down

We experimented with tuning these systems with the concatenation of WMT newstest sets from 2014-2018 yielding a tuning corpus of 14,820 parallel sentences. For each of the five separate transformer models trained for the Marian transformer-base ensemble systems in Gwinnup and Anderson (2020), continued training was performed for 10 epochs on the concatenated tests sets. An ensemble of the five resulting tuned models is then used to decode newstest sets from 2019-2021. Resulting scores reported by SacreBLEU are shown as Row 2 in Table 3, while the baseline, untuned ensemble is shown as Row 1. We note gains between +2.0 and +3.5 BLEU as measured by SacreBLEU over the baseline ensemble system depending on test set.

4 Experimental Results

Results reported here and in Table 3 for Marian systems were scored with SacreBLEU (Post, 2018) while results for OpenNMT systems were score with mult-bleu-detok.perl from the Moses toolkit (Koehn et al., 2007). Internal comparisons between the two scoring methods have been in agreement. All scores are on detokenized cased output.

The primary submission system was the OpenNMT-tf configuration described in section 3.1 and shown in Table 3 as onmt+asr-tune. It resulted in official scores of 53.31 BLEU-all, 38.83 BLEU-A, 39.56 BLEU-B, 0.64 chrf-all, 0.63 chrf-A, and 0.64 for chrf-B on the 2021 test-set.

Post evaluation a model with the OpenNMT-tf configuration described in section 3.1 was trained on all the unfiltered data (approx. 76M million bi-text). The results are shown in Table 3 as onmt-large. The baseline onmt-large system was approximately +1 BLEU better that the baseline onmt-asr system while the onmt-asr system which continued training with two epochs of the large data set and tuned with newstest2014-2017 (onmt-+asr-tune) was +2.5 BLEU better than the baseline onmt-large system which was trained with 10 epochs and comparable to the onmt-large system tuned with newstest2014-2017. Experiments were conducted on both onmt+asr and onmt-large with tuning sets comprised of different combinations of the supplied news test sets from 2014 to 2019. Tune7 is news test sets from 2014-2017 (11,820 bi-text), tune8 is news test sets from 2014-2018 (14,820 bi-text), and tune9 is news test sets from 2014-2019 (16,820 bi-text). Systems were tuned for three epochs using these tuning sets. Generally performance dropped off or decreased slightly with more than 3 epochs of tuning. To be consistent across systems and tuning sets we are only reporting results for 3 epochs. As can be seen in Table 3 all three tuning sets provided significant improvements over the baseline systems, generally in the range of +3.5 BLEU on test 2021. For onmt+asr there was little difference in tuning with tune7 or tune8 whereas tune9 was approximately +0.4 BLEU better than those two. For onmt-large tune7 did not provide as much benefit as tune8 and tune9 which were basically the same, less than 0.1 BLEU difference between the two.

5 Conclusion

While our two submission systems employ a standard method of fine-tuning to adapt models towards a test set, we find that our methods to sample a similarly-sized tuning corpus from a larger body of text while only using information about the source side of that data yields a reasonable improvement in translation quality. Such a technique could be useful in adapting translation models to specific domains where only the source language of a text source is available.

Using actual in-domain data, such as the provided news development sets, for fine-tuning provide a substantial gain in translation quality. Such data is not always available and thus other selection techniques as described in Section 2.3 come into play. Future work will investigate combining the two approaches to see if additional gains can be obtained.

The authors would like to thank Emily Conway and Braeden Bowen for their assistance in human evaluation of MT output.

References

2021. Findings of the 2021 conference on machine translation (WMT21). In *Proceedings of the Sixth Conference on Machine Translation*, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Grant Erdmann and Jeremy Gwinnup. 2019. Quality and coverage: The AFRL submission to the WMT19 parallel corpus filtering for low-resource conditions task. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 267–270, Florence, Italy. Association for Computational Linguistics.

Miquel Esplà, Mikel Forcada, Gema Ramírez-Sánchez, and Hieu Hoang. 2019. ParaCrawl: Web-scale parallel corpora for the languages of the EU. In *Proceedings of Machine Translation Summit XVII Volume 2: Translator, Project and User Tracks*, pages 118–119, Dublin, Ireland. European Association for Machine Translation.

Markus Freitag and Yaser Al-Onaizan. 2016. Fast Domain Adaptation for Neural Machine Translation. *CoRR*, abs/1612.06897.

Jeremy Gwinnup and Tim Anderson. 2020. The AFRL WMT20 news translation systems. In *Proceedings of the Fifth Conference on Machine Translation*, pages 207–212, Online. Association for Computational Linguistics.

Jeremy Gwinnup, Tim Anderson, Grant Erdmann, and Katherine Young. 2018. The AFRL WMT18 systems: Ensembling, continuation and combination. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 394–398. Association for Computational Linguistics.

Jeremy Gwinnup, Timothy Anderson, Grant Erdmann, Katherine Young, Michaeel Kazi, Elizabeth Salesky, Brian Thompson, and Jonathan Taylor. 2017. The AFRL-MITLL WMT17 systems: Old, new, borrowed, BLEU. In *Proceedings of the Second Conference on Machine Translation*, pages 303–309, Copenhagen, Denmark. Association for Computational Linguistics.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016a. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016b. Bag of tricks

Opinions, interpretations, conclusions and recommendations are those of the authors and are not necessarily endorsed by the United States Government. Cleared for public release on 23 August 2021. Originator reference number RH-21-122279. Case number AFRL-2021-2778.

		WMT newstest							
#	system name	2014	2015	2016	2017	2018	2019	2020	2021
1	marian-ens5-base	40.2	34.4	34.8	38.0	33.01	35.8	35.0	47.1
2	marian-ens5-tune	–	–	–	–	–	38.4	37.0	50.6
3	WMT20 onmt-base	36.87	32.58	32.48	35.50	30.76	38.26	–	–
4	WMT20 onmt-tune7	–	–	–	–	32.31	39.27	–	–
5	onmt+asr	–	–	–	–	33.17	38.08	35.86	51.05
6	onmt+asr-tune	–	–	–	–	35.71	40.39	37.61	54.49 (+3.44)
7	onmt+asr-tune7	–	–	–	–	36.15	40.91	37.54	54.58 (+3.54)
8	onmt+asr-tune8	–	–	–	–	–	40.72	37.67	54.72 (+3.67)
9	onmt+asr-tune9	–	–	–	–	–	–	38.04	55.08 (+4.03)
10	onmt-large	–	–	–	–	33.81	38.87	36.49	51.92
11	onmt-large-tune7	–	–	–	–	36.08	41.15	38.15	54.61 (+2.69)
12	onmt-large-tune8	–	–	–	–	–	40.90	38.40	55.48 (+3.56)
13	onmt-large-tune9	–	–	–	–	–	–	38.01	55.43 (+3.51)

Table 3: Experimental results for baseline and tuned systems. Marian systems are scored with SacreBLEU, OpenNMT-tf systems are scored with multi-bleu-detok.perl. Newstest2021 scored with the two supplied references. Systems 3 and 4 are WMT20 systems for progress comparison.

for efficient text classification. *arXiv preprint arXiv:1607.01759*.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. 2018. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Vincent Nguyen, Jean Senellart, and Alexander Rush. 2018. OpenNMT: Neural machine translation toolkit. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers)*, pages 177–184, New Orleans. Association for Machine Translation in the Americas.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Minh-Thang Luong and Christopher D. Manning. 2015. Stanford Neural Machine Translation Systems for Spoken Language Domain. In *International Workshop on Spoken Language Translation*, Da Nang, Vietnam.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *International Conference on Learning Representations (ICLR) Workshop*.

Brian Ore, Eric Hansen, Tim Anderson, and Jeremy Gwinnup. 2020. The AFRL IWSLT 2020 systems: Work-from-home edition. In *Proceedings of the 17th International Conference on Spoken Language Translation*, pages 103–108, Online. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Gema Ramírez-Sánchez, Jaume Zaragoza-Bernabeu, Marta Bañón, and Sergio Ortiz-Rojas. 2020. Bifixer and bicleaner: two open-source tools to clean your parallel data. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 291–298, Lisboa, Portugal. European Association for Machine Translation.

Víctor M. Sánchez-Cartagena, Marta Bañón, Sergio Ortiz-Rojas, and Gema Ramírez-Sánchez. 2018. Prompsit's submission to wmt 2018 parallel corpus filtering shared task. In *Proceedings of the Third Conference on Machine Translation, Volume 2: Shared Task Papers*, Brussels, Belgium. Association for Computational Linguistics.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2019. Wikimatrix: Mining 135m parallel sentences in 1620 language pairs from wikipedia. *CoRR*, abs/1907.05791.

Rico Sennrich, Alexandra Birch, Anna Currey, Ulrich Germann, Barry Haddow, Kenneth Heafield, Antonio Valerio Miceli Barone, and Philip Williams. 2017. The University of Edinburgh's neural MT systems for WMT17. In *Proceedings of the Second Conference on Machine Translation, Volume 2: Shared Task Papers*, pages 389–399, Copenhagen, Denmark. Association for Computational Linguistics.

Gurjeet Singh, Facundo Memoli, and Gunnar Carlsson. 2007. Topological Methods for the Analysis of High Dimensional Data Sets and 3D Object Recognition. In *Eurographics Symposium on Point-Based Graphics*. The Eurographics Association.

Jason R. Smith, Herve Saint-Amand, Magdalena Plamada, Philipp Koehn, Chris Callison-Burch, and Adam Lopez. 2013. Dirt cheap web-scale parallel text from the common crawl. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL '13)*, pages 1374–1383, Sofia, Bulgaria.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 6000–6010.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The united nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*.