All for One or One for All: Ensemble of Diverse Augmentation for Self-Attention

Stanford CS224N Default Project

Jasper McAvity
Department of Computer Science
Stanford University
jmcavity@stanford.edu

Tiffany Zhao
Department of Computer Science
Stanford University
tiffzhao@stanford.edu

Amir Zur
Department of Computer Science
Stanford University
amirzur@stanford.edu

Abstract

Accurate question and answering systems are crucial to web search engines that can tailor specifically towards user needs and information. Our project strives to produce a question answering system that works well specifically on the SQuAD 2.0 dataset. We extend the provided baseline BiDAF architecture through a character embedding layer and an implementation of self-attention inspired by QANet and R-NET. We further experiment with the data augmentation technique of backtranslation using a variety of languages, and seek to answer the question: is ensembling many small models, each augmented by a unique language, better than training a single large model on the full augmented dataset? Our final model, an ensemble of our self-attention model and a BiDAF model trained augmented data, achieves an **EM** score of **62.93** and an **F1** score of **65.78** on the held out test set, and significantly improves upon the baseline BiDAF model on the development set.

1 Key Information to include

- Mentor: Kendrick Shen
- External Collaborators (if you have any): None.
- Sharing project: No.

2 Introduction

Accurate question and answering systems are crucial to web search engines that can tailor specifically towards user needs and information. Our project strives to produce a question answering system that works well specifically on the SQuAD 2.0 dataset. In the question-answering task, a model is presented a context paragraph extracted from Wikipedia and a question; the model is then asked to produce an answer to that question. The question is *answerable* if its answer is contained within the context paragraph – in this case, the model can generate the answer by outputting the starting and ending indices of the answer within the context. However, about half of the questions are *unanswerable* given information from the context alone – in this case, the model must output a special *no answer* response.

We aim to produce a model which outperforms the baseline BiDAF model. The key component of the BiDAF model is its query-to-context attention, which allows it to internally query its encoding of the context paragraph via its encoding of the question, in order to extract the relevant information

Question: Why was Tesla returned to Gospic?
Context paragraph: On 24 March 1879, Tesla was returned to Gospic under police guard for not having a residence permit.
Answer: not having a residence permit.

Figure 1: Example of an input-output pair for the SQuAD question-answering task. The inputs consist of the Question and Context paragraph, and the model is tasked with producing the Answer.

for producing the final answer. The method of attention has gained recent success in the form of self-attention [1], leading to a class of high-performing Transformer-based models. In our project, we extend the BiDAF model architecture by experimenting with different methods of self-attention, in order to achieve greater accuracy on the SQuAD question-answering task.

One strong feature of self-attention is that through this layer, a model trains itself to extract the most useful information from its current state. This means that a model built on self-attention has the potential to generalize without quickly overfitting, and benefits from large amounts of training data. Hence, methods of data augmentation are useful in creating high-performing models. One technique used for the Question-Answering task is backtranslation [2], which augments data by using an existing Neural Machine Translation (NMT) model to translate a question from the SQuAD dataset into another language, and then back into English. This achieves the goal of rephrasing the question, while keeping its original meaning. In our paper, we seek to extend the backtranslation method of [2], which uses German in order to rephrase its questions, by experimenting with a variety of diverse language families. We hypothesize that languages which are further away from English, measured by the BLEU score of the NMT translation, will create more diverse training data and thus lead to a more accurate question-answering model.

One simple yet powerful technique used in machine learning is ensembling [3], which combines the predictions of many smaller models in order to construct a high-performing language model. In our paper, we ensemble a variety of question-answer models each trained on a separate language. We seek to answer the following question: does one large model, trained on augmented data pooling together backtranslation from different languages, perform better than an ensemble of smaller models, each trained on augmented data that is backtranslated from a different language?

3 Related Work

One successful model architecture for the SQuAD question-answering task is the BiDAF model [4], which employs the technique of attention to map question encodings to answers within the context encodings. Since then, other model techniques have used the concept of self-attention [1] in order to boost model performance. This includes the R-NET model [5], which introduces a Self-Matching Layer, and the QANet model [2], which uses a self-attention layer within its Encoder block in a similar manner to the Transformer model [1]. In our paper, we implement self-attention using the QANet architecture and the Self-Matching Layer of the R-NET architecture.

Other models have achieved high performance on the SQuAD task through the use of external features and data augmentation [6]. For example, the QANet model augments the SQuAD data translating questions from English to German and then back to English, which rephrases the original question. This method is known as backtranslation [2]. In our paper, we experiment with backtranslation across a variety of languages. Many models also implement ensembling techniques [3], whereby multiple multiple models, trained on different splits of the training data, are pooled together in order to create a stronger model. In our paper, we use an ensembling technique known as Bayesian voting, which takes a weighted sum of the model predictions based on an assignment of priors for each model.

4 Approach

Our approach consists of implementing two different neural network architectures: the QANet model [2] and the Self-Matching layer of the R-NET model [5]. Both architectures further improve on the baseline model by implementing not only GloVe word embedding [7], but also character embedding

using a convolutional neural network, as used in the original BiDAF model [4]. We train our model on augmented data through backtranslation, and lastly pool together different model architectures through ensembling.

4.1 QANet

Neural network models have achieved recent success through the use of self-attention [1], through which a model infers the most useful information from its own hidden states. The QANet model [2] adapts the use of self-attention in the Transformer model [1] to the question-answering task.

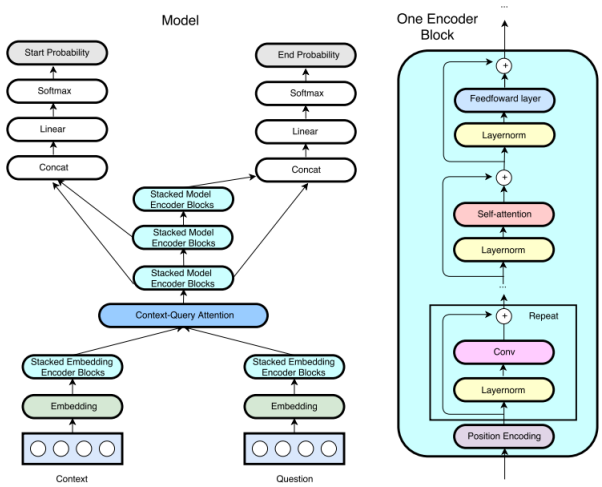

Figure 2: Outline of the QANet model architecture. Its main unit is the Encoder block, which uses a self-attention layer.

The architecture of the QANet model is shown in Figure 2. It follows a similar flow as the BiDAF model [4]: an embedding layer of the input text, an encoding of the embedding layer, a context-to-query attention layer, an encoding of the attention layer, and finally an output layer which produces the p_{start} and p_{end} distributions over the start and end indices of the final answer. The main unit of the QANet model is the Encoder block, which is divided into three parts: a convolutional layer, a self-attention layer, and a feed-forward layer. Each sub-layer of the Encoder block is also wrapped within a residual layer, so that the loss effectively back-propagates across all weights during training.

We implemented the QANet ourselves using the Pytorch library. As noted in the paper, we use depthwise-separable convolution [8] in order to save on parameter space. Our positional encoding layer within the Encoder block makes use of alternating sinusoidal perturbation of the word indices, as done in the original Transformer model [1].

4.2 Self-Attention and CoAttention

R-Net is another model architecture that utilizes self-attention. The model is similar to BiDAF, with embeddings, encoder, and context to query/query to context layers, but additionally adds a self-attention layer after that along with attention to the initial query vector when producing output.

Another type of attention is CoAttention, which involves second-level attention, which is attending to attention outputs in a single CoAttention layer.

We implemented R-Net and a CoAttention layer ourselves using Pytorch. We made use of a general attention-pooling layer to perform most of the attention computations in R-Net. We experiment with replacing layers in the baseline model with a CoAttention layer as well as adding R-net's self-attention layer to the baseline. We found that CoAttention didn't improve on the baseline and that R-net actually did worse. This second result was surprising, as R-net is a sophisticated model, which points to a likely implementation error. However, using a BiDAF model with a self-attention layer added in improved on the baseline. These results will be shown later in the paper.

4.3 Data Augmentation

We utilize the MarianMT Tokenizer and Neural Machine Translation model to perform data augmentation [9]. Using the transformer from Hugging Face, we coded up ourselves to use this framework to translate only the questions from the dataset into another language then back to English. We augmented 10,000 random sequences from the training dataset using backtranslation for each of the following languages: Arabic, Chinese, French, Finnish, Italian, Russian and Spanish with an example shown in Table 1.

Original: *What general species of animal was the marine reserve designed to protect?*

Language	Translation
Arabic	What generic species of animals have marine reserves been designed to protect them?
Chinese	What are the general animal species protected by marine protected areas?
French	What general animal species has the Marine Reserve been designed to protect?
Finnish	What common species of animals was the marine area designed to protect?
Italian	What general species of animal was the marine reserve intended to protect?
Russian	What general species of animals were intended to protect the marine reserve?
Spanish	What general animal species was the marine reserve designed to protect?

Table 1: An example of backtranslation results of various languages.

We then calculated the average BLEU score across all 10,000 generated sequences for each language. We used the nltk package and a smoothing function for shorter translations to measure each language's backtranslation model performance.

Language	Average BLEU Score
Arabic	0.127
Chinese	0.029
French	0.255
Finnish	0.320
Italian	0.761
Russian	0.320
Spanish	0.646

Table 2: Comparison of the average BLEU score for each language across all generated sequences.

We can see that Italian and Spanish have the highest BLEU scores for this particular translation which makes sense due to the similarities in Italian and Spanish linguistics with the English language. This discovery led us to hypothesize utilizing Italian and Spanish augmented data could further improve our question and answering system's performance. On the other hand, the Chinese and Arabic languages differ significantly from the English language due to the lower BLEU scores so we can expect a worse performance when augmented our dataset using these languages. This is consistent with the fact that Chinese and Arabic are part of the Sino-Tibetan family of languages, while English, French, Finnish, Italian, Russian, and Spanish are all part of the Indo-European family of languages as shown in Figure 3.

4.4 Ensemble Model

During testing, we pool the predictions of multiple models in order to construct an ensemble model with greater accuracy. When pooling predictions, we use Bayesian voting with equal priors [3], meaning that we average the start index distribution p_{start} and the end index distribution p_{end} across all smaller models. We then choose the p_{start}, p_{end} tuple which maximizes the combined prediction probability using dynamic programming. In our development process, we also experimented with instead taking the maximum probability score for each start and end index; however, we found that this approach consistently results in weaker predictions than Bayesian voting.

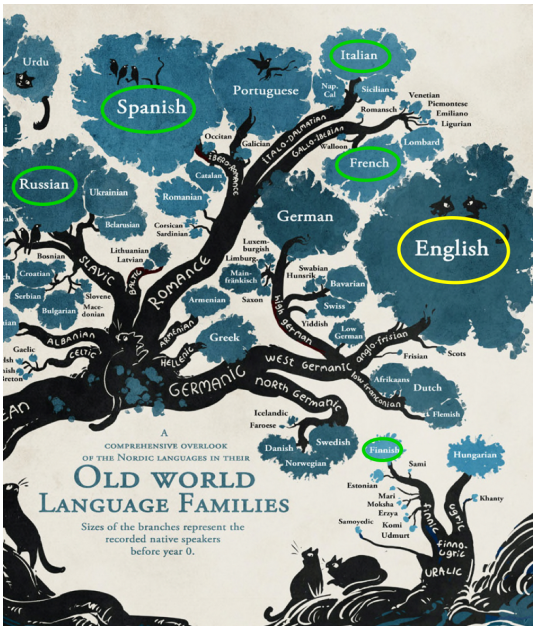

Figure 3: Part of the Indo-European Language Tree taken from Minna Sundberg in her web comic *Stand Still, Stay Silent*.

5 Experiments

Our experiments focus on the following research goals:

1. Construct a self-attention model achieving high accuracy on the SQuAD 2.0 question-answering task.
2. Compare the effect of backtranslation using different language families on overall model accuracy.
3. Compare a singular model trained on large augmented data to the ensemble of multiple models trained on small augmented data.

In this section, we will report on the first research goal, since our paper is focused on the SQuAD 2.0 question-answering task. We will report our results for the second two research questions in our Analysis section. The model with the highest level of accuracy constructed by our team is an ensemble of our extension to the BiDAF model with self-attention, our extension to the BiDAF model with character embeddings, and our extension to the BiDAF model with character embeddings, trained on our full augmented data.

5.1 Data

We use the official SQuAD 2.0 dataset as well as some new SQuaD 2.0 examples produced by the teaching team. Our training, development, and testing sets is split as follows:

- *train* (129, 941 examples): all taken from the official SQuAD 2.0 training set.
- *dev* (6078 examples): roughly half of the official dev set, randomly selected.
- *test* (5915 examples): the remaining examples from the official dev set, plus hand-labeled examples.

We further augment our data using backtranslation from Arabic, Chinese, French, Finnish, Italian, Russian, and Spanish. Each backtranslation augmentation added an additional 10,000 training datapoints. We will refer to the data augmented with all seven languages as the full augmented data.

5.2 Evaluation method

We evaluate our model performance using Exact Match (EM), F1, and AvNA scores. The EM score measures the percentage of answer that the model produces which agree one-to-one with the gold label answers. The F1 score is a relaxation of the EM score, which averages false positives (words in the model answer which are not present in the gold label) and false negatives (words in the gold label answer which are not present in the model answer). Lastly, the AvNA metric measures the percentage of questions for which the model correctly chooses whether or not to produce an answer. All metrics range between 0 and 100%, where higher scores correspond to better models.

5.3 Experimental details

Our highest-performing model is an ensemble of three different models. The first model (BiDAF self-attention) is an extension to the BiDAF model with the original architecture and parameters, with self-attention applied to the Context-to-Query attention layer. The second two models are an implementation of the BiDAF model with the original architecture and parameters. The BiDAF self-attention model and the first of the two BiDAF models (BiDAF char-embedding) are trained on the original training data. The last BiDAF model (BiDAF augmented) is trained on the full augmented data.

All models implement both 300-dimension GloVe word embedding and character level embedding, with character vectors of dimension 64 and an output embedding of dimension 200. The full embedding is created by concatenating the word and character embedding outputs and passing them through a highway network [10] with an output dimension of 100. All models are trained for 20 epochs using AdaDelta optimization with learning rate 0.5 and no L2 regularization [11], as in the original BiDAF paper. Lastly, all models are ensembled to construct one large model (BiDAF ensemble).

5.4 Results

Below are the performance results for each individual model, and the final ensembled model. We compare this model to the provided BiDAF baseline, which implements BiDAF without character embedding.

Model	F1	EM	AvNA
BiDAF baseline	61.17	57.65	68.14
BiDAF self-attention	63.28	60.21	69.12
BiDAF char-embedding	65.11	61.86	71.38
BiDAF augmented	63.22	59.92	69.53
BiDAF ensemble	**66.65**	**64.01**	**71.33**

Table 3: Model performance, in comparison with baseline BiDAF model. The highest performance is achieved by an ensemble of all BiDAF models excluding the baseline.

On the held-out test set, our ensemble model achieved an **EM** score of **62.93** and an **F1** score of **65.78**, placing in the top 20 model submissions as of Sunday, March 13th. This is a significant improvement over the baseline model, which shows the benefit of self-attention, data augmentation, and model ensembling. In our analysis section, we will discuss the effect of each of these extensions separately, focusing on our two other research questions.

6 Analysis

Below we present our analysis on the different components of our model. Specifically, we wish to answer the research questions presented in the Experiments section: which languages are most useful for backtranslation, and what is the best method of pooling together the diverse data that these languages help generate?

For this analysis, we consider the models generated by QANet [2]. Although our QANet achieved overall lower performance, as shown in Table 4, we choose this model for experimenting with

backtranslation because of the backtranslation used in its original paper. Our QANet uses the original parameters of the QANet paper, with the exception of 3 convolutional neural network layers with the Embedding Encoder instead of 4, and 4 Embedding Encoder blocks instead of 7. We train each of the QANet models discussed below for 10 epochs with the Adam optimizer, using $\beta_0 = 0.8$, $\beta_1 = 0.999$, $\varepsilon = 10^{-7}$, a learning rate of 0.001, and L2 weight decay of 3×10^{-7}.

Model Architecture	EM	F1	AvNA
BiDAF baseline	61.17	57.65	68.14
QANet	64.16	56.91	64.16

Table 4: Model performance of the QANet, in comparison with baseline BiDAF model.

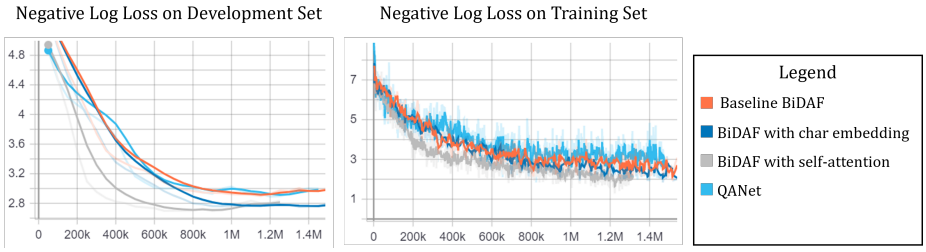

Figure 4: Model performance over multiple steps as shown by loss on the dev set (left) and loss on the train set (right). It is interesting to note that different model architectures yielded different loss patterns. For example, the QANet model had much more volatile training loss than other models, while the self-attention model converged earliest, and around the same time for both the dev and train set.

6.1 Backtranslating Languages

In this section, we seek to answer the following question: which language is most effective for backtranslation in the SQuAD question-answering task?

We hypothesized that languages which are further away from English, as measured by BLEU score in Table 2, will produce more valuable training points, since they will create more varied rephrasings of the original questions. To our surprise, we discovered the opposite effect: languages which are closer to English, such as French and Italian (which share Latin roots), lead to stronger models than languages which are further away from English, such as Arabic and Chinese. This is reflected below in Figure 5.

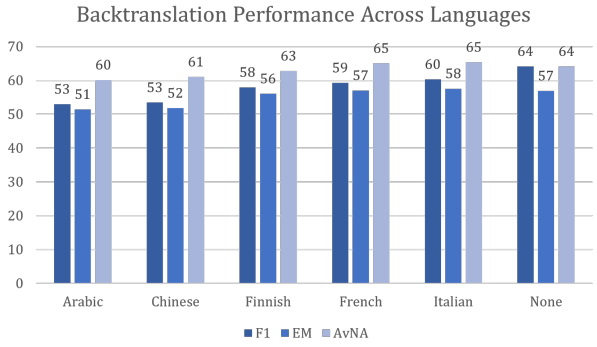

Figure 5: Comparison of model performance across different languages used for backtranslation augmentation. Note that not all languages improved upon the QANet model trained on the original data, None.

This finding agrees with the decision of [2] to use German, which is within the same language family as English, for its backtranslation. We are curious in a further syntactic exploration of the languages chosen in order to determine which re-phrasing of the question leads to the best model results.

6.2 Ensembling versus Augmenting

Our last key research goal is to determine which construction is optimal for a question-answering model: a model trained on augmented data from many different languages, or an ensemble of many models each trained on a unique language. We hypothesize that an ensemble of smaller models performs better than one large model – all for one beats one for all.

To test this theory, we train one large QANet model for 10 epochs on all 70,000 tokens generated by backtranslation of the seven languages Arabic, Chinese, Finnish, French, Italian, Russian, and Spanish. Unlike our smaller QANet models, this model uses all 4 convolutional neural layers in its Embedding Encoder Block, as in the original QANet paper [2]. We then construct an ensemble of all language models, whose individual performances are illustrated in Figure 5.

Model	F1	EM	AvNA
Single augmented model	55.1	52.85	61.33
Ensemble of language models	55.76	55.15	57.2

Table 5: Comparison in performance of a single model trained on the full augmented dataset, versus an ensemble model of models each trained on data augmented by a unique language.

We find that an ensemble model indeed does outperform a large model trained on all data, as shown in Table 5. In particular, its EM and F1 scores see an improvement. It is interesting to note the opposite effect on the AvNA metric – we hypothesize that this is because by averaging together distributions on starting and ending indices separately, we encourage an ensemble to produce answers more frequently than any of the smaller models within the ensemble.

We also note that the overall performance of the ensemble model is lower than the individual performance of the models. This is because by training our models on augmented data, we re-index their vocabulary. This requires a separate, specialized evaluation file for each of the small models within the ensemble that is re-indexed to match the augmented data. However, we tested the ensemble model on a single evaluation file, which the individual models also perform poorly on. In future work, we would like to construct a stronger ensemble by re-indexing each datapoint separately for each of the ensembled models. Nevertheless, even under this constraint, the ensemble model achieved better performance, hence corroborating our hypothesis that an ensemble of languages is better than a single model that is trained on all languages simultaneously.

6.3 Qualitative Results Analysis

With predictions from our best model, we divided questions by their question word to find out which question words our model performed best on.

Question Word	whose	whom	who	what	**when**	where	why	how	which
Percent Accuracy	60.0	71.4	77.8	68.2	**83.6**	59.3	52.6	71.2	66.9

Table 6: Model performance on different types of question words.

We see that the model performs best on 'when' questions, which makes intuitive sense, as the answers to these questions will often be numbers or specific words like months of days of the week that would make them easier to learn. In contrast, the model performed most poorly on 'why' questions. This result also makes sense - answering a 'why' question requires understanding both a thing that happened and the reason it happens, a much more difficult task than finding a date.

These results point to areas of improvement, but even more information about where the model could have gone wrong can be gained by looking at specific questions that it predicted incorrectly. Reading through these questions and answers reveals three broad categories of error: correct but too much, correct but not enough, and incorrect. The first two categories are very close to the correct answer, but

give too much or not enough context. In the example ('How does the pathogen kill the phagocyte?', 'digestive enzymes', 'by the activity of digestive enzymes'), where the middle is the correct answer and the right is the prediction, the model gives too much context, whereas in ("Where did one of Triton's daughters decide she wanted to hang out and stay?", 'coast of Denmark', 'Denmark'), it gives not enough. Something to note about these types of errors is that the problem with the model isn't understanding the question or even necessarily finding the answer, but determining how to phrase an answer correctly. Often, these types of errors give what is essentially the correct answer, so it is incorrect errors that are more concerning. In an example like ('How did Yesun Temur die', 'Shangdu', 'poison'), the model mistakes this 'how' for a 'where', causing it to give the wrong answer. In other examples, like ('Who owned the Cuckoo Tavern?', 'Paul Revere', 'Jack Jouett'), the model understands the question but gives the wrong answer. These results show that to improve the model, one could either take steps to reinforce understanding of question, which might involve more query focused attention, or attempt to increase understanding of the passage, which could involve more passage focused attention.

Taken together, looking deeper into the specific questions that were missed shows that our model actually performs better than our other metrics might indicate. The presence of a significant number of the first two types of error show that it was actually able to find acceptable or close to acceptable answers even when it was marked as incorrect.

7 Conclusion

In conclusion, we find that self-attention is a powerful technique for the SQuAD question-answering task, offering significant improvement upon the baseline BiDAF model. We further find that self-attention benefits from augmented data, generated by backtranslation from languages which are linguistically similar to English. Lastly, we assert that an ensembling of diverse models yields better performance than the construction of a singular large model, and use ensembling in conjunction with data augmentation and self-attention to achieve significant improvement upon the baseline model.

Our work is limited by the amount of time and resources our machine learning models could use in training. In future work, we would like to make the QANet model more robust, and test it on backtranslated data from a variety of language families. An interesting direction for future research could be syntactically analyzing which type of question rephrasing is most beneficial for the QANet on the SQuAD 2.0 question-answering task.

References

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.

[2] Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. Qanet: Combining local convolution with global self-attention for reading comprehension, 2018.

[3] Thomas G. Dietterich. Ensemble methods in machine learning, 2000.

[4] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension, 2018.

[5] Natural Language Computing Group. R-net: Machine reading comprehension with self-matching networks. May 2017.

[6] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[7] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[8] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[9] 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. Marian: Fast neural machine translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[10] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Highway networks. *arXiv preprint arXiv:1505.00387*, 2015.

[11] Matthew D. Zeiler. ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701, 2012.