Seq2Seq Group Project - Cold and Dynamic Fusion

Lisa Kuhn **Philipp Meier**
Departement of Computational Linguistics
University Heidelberg
{kuhn|meier}@cl.uni-heidelberg.de

Abstract

In fusion methods a translation model uses a language model to generate a translation. Incorporating language models into traditional neural translation models allows the model to overcome possible domain gaps during decoding and to increase fluency. This also results in an increase of available data for low resource language pairs. In this project we implemented Cold and Dynamic fusion as well as two additional variants of Cold Fusion. Finally, we compared the models' BLEU scores and analyzed the models' output.

1 Introduction

Fusion methods in machine translation incorporate a language model into a translation model to improve downstream performance. There are a number of advantages to using a language model in this way. As high-quality parallel corpora are not available for every language pair, it seems practical to leverage knowledge from high-quality monolingual corpora, for example in the form of a language model. Consequently the language model is able to improve fluency of the generated text of the Seq2Seq model.
There are different methods to integrate a language model into an RNN translation model. In early approaches both models were trained separately before the actual integration.

Such early methods like Shallow Fusion only use the language model during inference, in which the language model proposes a set of candidate words at each time step t. Possible translation candidates are then scored according to the sum of the scores given by the language Model and the translation model as shown in formula 1.

$$\log p(y_t = k) = \log p_{TM}(y_t = k) + \beta \log p_{LM}(y_t = k) \quad (1)$$

Another method, called Deep Fusion (Gulcehre et al., 2015), deploys a stronger connection between the Seq2Seq model and the language model. To achieve this, the language model's and the translation model's hidden states are concatenated and along with the previous word and the context vector, they are used to compute the next word. As formula 2 shows, this fusion is more advanced.

$$p(y_t|y_{<t}, x) \propto exp(y_t^\top (W_\circ f_\circ(s_t^{LM}, s_t^{TM}, y_{t-1}, c_t) + b_\circ)) \quad (2)$$

In order to balance the language model's influence, a gating mechanism similar to formula 3 is used. This procedure produces a scalar which denotes the importance of the language model. This gating is enhanced in Cold Fusion.

$$g_t = \sigma(v_g^T s_t^{LM} + b_g) \quad (3)$$

2 Methods

This section gives an overview of Cold and Dynamic Fusion.

2.1 Cold Fusion

Cold Fusion (Sriram et al., 2017) differs from the previous methods as a Seq2Seq model is trained together from scratch with a pre-trained and fixed language model. Therefore Cold Fusion is an 'early training integration' approach. Formula 4 shows the computation of h_t^{LM} which is generated by feeding the logit output of the language model into a deep neural network. h_t^{LM} is then used to compute the gating value g_t. Unlike in Deep Fusion, Cold Fusion employs a fine-grained gating mechanism. This means a vector is computed instead of a scalar as in Deep Fusion. The final fused state is produced by concatenating s_t and the Hadamard product of the gating value and h_t^{LM}. The final

output is then generated via a deep neural network (see formula 7) and used to predict the next word in formula 8.

$$h_t^{LM} = DNN(\ell_t^{LM}) \quad (4)$$

$$g_t = \sigma(W[s_t; h_t^{LM}] + b) \quad (5)$$

$$s_t^{CF} = [s_t; g_t \circ h_t^{LM}] \quad (6)$$

$$r_t^{CF} = DNN(s_t^{CF}) \quad (7)$$

$$\hat{P}(y_t|x, y_{<t}) = softmax(r_t^{CF}) \quad (8)$$

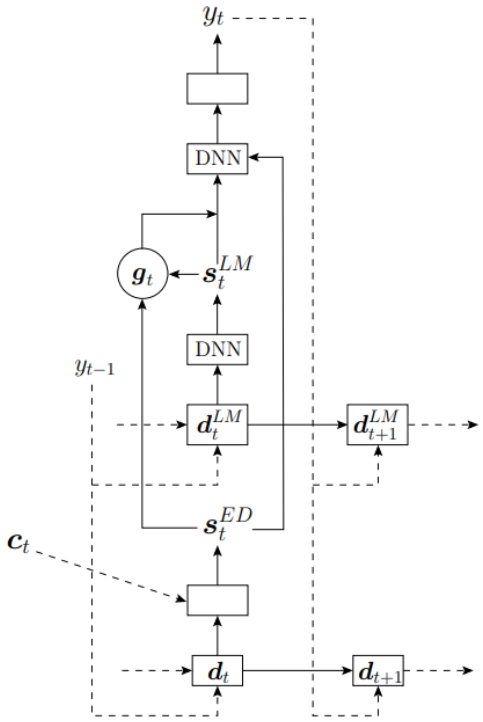

Figure 1: Cold Fusion mechanism (Toshniwal et al., 2018)

2.2 Dynamic Fusion

Dynamic Fusion (Kurosawa and Komachi, 2019) uses an attention architecture to integrate the language model into the translation model. The name Dynamic Fusion is inspired by the fact that the fusion of the translation model and the language model happens in the attention module of the Seq2Seq model. In this module, dynamic weights are used instead of fixed weights as in previous approaches.

$$\alpha_{word} = \frac{exp(e_{word}^T S_{TM}(y|x)}{\sum_{word \in V} exp(e_{word}^T S_{TM}(y|x))} \quad (9)$$

$$c_{word} = \alpha_{word} e_{word} \quad (10)$$

$$c_{LM} = \sum_{word} c_{word} * P_{LM}(\mathbf{y}; y = word) \quad (11)$$

$$h_{TM} = [S_{TM}(\mathbf{y}|\mathbf{x}); c_{LM}] \quad (12)$$

$$S_{ATTN} = W h_{TM} \quad (13)$$

$$\hat{y} = \arg\max_y softmax(S_{ATTN}) \quad (14)$$

Unlike in the previously presented approaches, it is not necessary to use identical vocabularies for the language and the translation model, because the language model's probability of the corresponding word is multiplied with a word attention in formula 11 and not concatenated with the translation model's hidden state. First, the alignment score α_{word} is computed, which is used to calculate the word attention c_LM. The language model comes into play in formula 11, which is where the language model's probability is used. Finally, the hidden state of the translation model S_{TM} and the word attention c_{LM} are concatenated to produce the hidden state h_{TM}. By multiplying a weight matrix and h_{TM} we receive S_{ATTN}, which is then used to predict the final target word $\hat{y}$. Figure 2 shows the Dynamic Fusion mechanism.

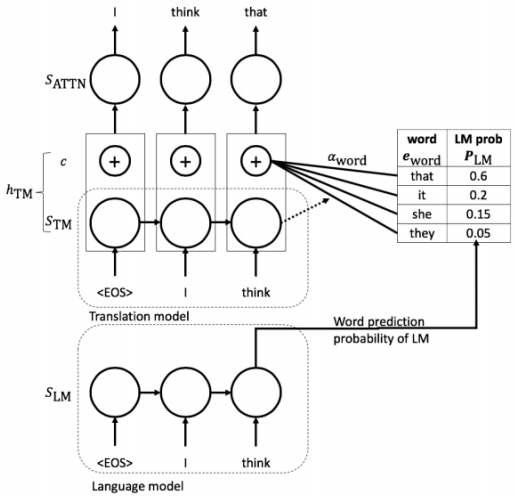

Figure 2: Dynamic Fusion mechanism (Kurosawa and Komachi, 2019)

2.3 Data

Translation Data: As parallel data, we decided to use the Europarl Corpus (Koehn, 2005). The

Europarl corpus is sentence-aligned and contains proceedings of the European Parliament. In order to determine the size of our training, validation and test data we used the corpus details from (Kurosawa and Komachi, 2019). We used one million sentences as training data for the translation model. For validation and test data 3500 sentences each were used. Additionally, we introduced two length constraints, only sentences with a maximum of 60 words were considered and the length difference between the sentences in a sentence pair had to be smaller than 15 words.

2.4 Baseline

The baseline model was used for comparison and later for language model integration. This attentional Seq2Seq model consists of a two-layered Bi-LSTM encoder with 1024 hidden units. The decoder consists of a two-layered LSTM decoder with 1024 hidden units. To implement the baseline and the Fusion methods, we chose OpenNMT-py (Klein et al., 2017) as our framework.

2.5 Language Model

Since we observed poor results with the language model trained on Gigaword, we deployed an additional language model trained on the monolingual English data of the Europarl corpus. The language models consist of a three-layered GRU with a hidden size of 1024 and an embedding size of 512. They were trained using the Adam Optimizer, a batch size of 128 and an initial learning rate of 0.0005 for the Gigaword language model and 0.001 for the Europarl language model.

Language Model Data: The data used for the Gigaword language model were 2,000,000 sentences (which is the sentence count of the LM in Dynamic Fusion) of the Gigaword-5 corpus and the data for the Europarl language model were the monolingual english sentences of the Europarl corpus. The Gigaword language model achieved a final test perplexity of 96.0, the Europarl language model a final test perplexity of 56.67. Both language models shared the same vocabulary as the translation model. Additionally, we used byte-pair encoding

3 Results

For the evaluation a beam size of 5 was used. Since we deployed two language models, we made multiple runs using different language models.

Model	Bleu Score
Baseline	26.77
Baseline + Cold Fusion	16.76
Baseline + Cold Fusion Variant 1	18.02
Baseline + Cold Fusion Variant 2	-
Baseline + Dynamic Fusion	23.32

Table 1: Results on the evaluation dataset with Gigaword Language Model

Model	Bleu Score
Baseline	24.64
Baseline + Cold Fusion	22.03
Baseline + Cold Fusion Variant 1	21.16
Baseline + Cold Fusion Variant 2	0
Baseline + Dynamic Fusion	23.45

Table 2: Results on the evaluation dataset with Europarl Language Model

The results were surprising, especially for the models using the Gigaword language model, because our models were not able to surpass the baseline model. In our experiments, we deployed a cold fusion model called 'variant 1' which applied the gating in formula 5 to the hidden state of the translation model instead of the hidden state of the language model. The results show, that when using the Gigaword language model, the model is able to surpass the vanilla Cold Fusion. However, in the second run, variant 1 had a lower BLEU score than vanilla Cold Fusion. The performance difference between both variant 1 models can be explained as follows: Since we apply the gating to the translation models' hidden state instead of the hidden state of the language model, the variant 1 which uses the Gigaword language model is able to improve its performance, because the influence of the language model is limited. The opposite is the case in our second run, where the Europarl language model is used: The Europarl language model seems to have a positive influence on the translation and limiting this influence of the language model harms the performance.

Variant 2 uses the gating (g) to compute the logit directly. To do this, s_t^{CF} was not computed (Equation 6) and replaced by g in equation 7. In both runs, this variant was not able to produce

reasonable results and achieved zero BLEU score.

Dynamic Fusion was able to achieve the second best results. As can be seen in table 1 and 2, Dynamic Fusion is only able to gain a small advantage from the Europarl language model with an improvement of 0.12 BLEU. Dynamic Fusion uses the language model as auxiliary information, which could be the reason for the better results and higher robustness. We deployed a case study in section 4.2.

4 Analysis

We deployed a linguistic analysis of the generated outputs and an analysis of the training performance. In these analyses, we focus on the results generated by the Europarl language model.

4.1 Training

Accuracy: Figure 3 shows that the baseline yields the highest validation accuracy. Dynamic Fusion is relatively close to the baseline and both reach their maximum after around 30 000 epochs. However, Cold Fusion and its variant start with a lower validation accuracy of 21% and respectively 32%. The validation accuracy of Cold Fusion approximates the variant 1 accuracy and surpasses it after 45 000 epochs. After 100 000 epochs, their validation accuracy scores are close to each other with 51% and respectively 49%. The difference between these graphs is interesting: While the Cold Fusion graphs start with a low validation accuracy and improve with time, the Dynamic Fusion graph starts with a higher validation accuracy, but does not improve much compared to the graphs of Cold Fusion. One of the reasons for that is that the Cold Fusion had to be trained with a smaller but constant learning rate than the other models, because in previous runs the models' accuracy and perplexity increased during training, meaning the model failed to find the minimum.

Perplexity: A similar difference arises when plotting the perplexity in figure 4. The baseline and Dynamic Fusion have a validation perplexity of around 6.0 to 7.0 when starting training, while Cold Fusion and its variant first have to decrease the perplexity from around 1500 respectively 120. Finally, all models are able to achieve a low validation perplexity. Plots of the models using the Gigaword language model are provided in the appendix.

4.2 Case study

Baseline vs. Cold Fusion In an overview of the output of Cold Fusion, we can see that the model sometimes produces repetitions. However, there are also interesting phenomena generated by the model. For instance this sentence below, which is rather simple but the models seem to have problems to translate it correctly.

> they have become second-class citizens who are no longer able to express themselves publicly, and afghanistan was once a progressive country!

The baseline outputs: *they have become the second class people who can no longer speak out in public, and afghanistan was once more an automatic.* Cold Fusion generates: *they have become second-class citizens who cannot speak in the public and afghanistan was once again a subject that was made in public.* In these sentences both the baseline and Cold Fusion seem to have problems. While Cold Fusion seems to stick more to the reference sentence, it outputs wrong words at the end 'again a subject that was in public'.

We can see that the language model in Cold Fusion can cause an alternative phrasing:

> in my opinion, this last point is particularly important and could be especially useful in protecting the community's financial interests.

The baseline produces: *my last point is of special importance, in particular in the protection of the community's financial interests.* Cold Fusion outputs: *i believe that the last point is particularly important - especially to protect the financial interests of the community.* Cold Fusion introduces a hyphen and generates *'financial interests of the community'* instead of *'community's financial interests'*

It is also worth mentioning, that sometimes the baseline produces repetitions, while Cold Fusion does not have this problem with the same sentence:

The baseline outputs: *i believe that baroness ashton, the decision by baroness ashton to send a european union delegation is an important gesture of support and i would like to thank her for having*

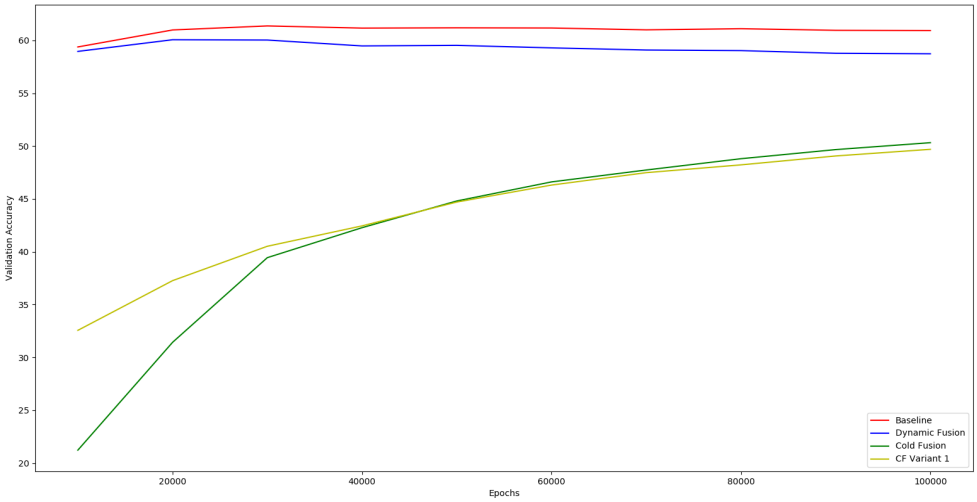

Figure 3: Validation accuracy

considered me with the presidency.. We can see a repetition of *baroness ashton*. Cold Fusion outputs: *i believe that the decision of baroness ashton, a delegation of the european union is to send an important gesture of support, and i would like to thank her for her attention to the chair.*

The influence of the language model is also shown here:

> in the netherlands, this figure was even as high as 85%, and in the netherlands, unlike in france, where only the president decides on such things, it is parliament that decides on the subject of a referendum.

The baseline outputs: *in the netherlands, this figure was even 85 % and, unlike france, where it is only the president who decides on such decisions, the netherlands has decided to hold parliament through referenda.* Cold Fusion produces: *in the netherlands, this figure is even 85 % and contrary to france, where only the president of the president of these decisions will not be held in the netherlands, the parliament will be holding referendums.* In this output we can see the influence of the language model. *Unlike* is replaced with *contrary*. Unfortunately, the translation contains repetitions caused by the language model again.

Baseline vs. Dynamic Fusion If we take a look at the generated output, we can see that Dynamic Fusion suffers from repetitions, but it is also able to enrich the output. We can also see that sometimes the Dynamic Fusion translation is closer to the reference. Consider this reference sentence:

> i trust the commission took careful note of events in the house this morning during the vote on the budget.

The baseline outputs: *i assume that the commission has noted exactly what has happened in plenary this morning during the vote on the budget.* Dynamic Fusion outputs: *i imagine that the commission has taken careful note of what happened in the house this morning when it voted on the budget vote this morning.*.
It can be seen that the language model proposes 'imagine' instead of 'assume' or 'trust' and the part 'has taken careful note' is also closer to the reference translation. However, two issues can be found: Firstly, the repetition of 'this morning' and secondly the usage of 'it'. This 'it' changes the meaning of the sentence, it suggests that the commission has voted on the budget vote.
Unfortunately, BLEU measures the overlap of the reference sentence with the model's output. However, if a language model proposes synonyms, which we have seen in the examples, and improves the general translation, BLEU will not take this into account. This means outputs which include alternative words are not rewarded.
The negative influence of the language model for Dynamic Fusion can be seen when looking at this

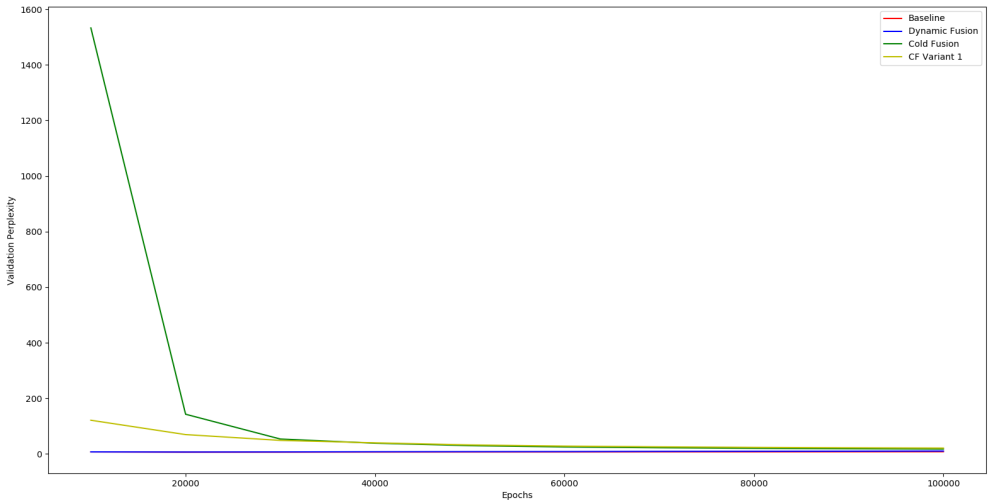

Figure 4: Validation Perplexity

reference sentence:

> this was indeed a tragedy of enormous proportions, with significant and perhaps enduring consequences, but the amount of deliberate spills from oil tankers along all our coasts dwarfs the amount of oil that was spilled from erika.

The baseline outputs: *this was indeed a tragedy which caused significant and potentially unwarranted damage, but the perpetrators of our coastlines deliberately became schemesitories to the sea: the owners of the sea at the sea.* The output of dynamic fusion is: *this cheque was, in fact, a dramatic inventory, which has done considerable damage to the potentially exposed and potentially exceptional damage, but the czechs are intentionally turning the shots on the sea into the sealing off with restrictions.* We can see that while baseline also struggles with this sentence, the Dynamic Fusion model fails completely to capture the meaning and generates nonsense.

Nevertheless the language model also has positive effects. Taking this reference sentence:

> but it has been nine months since the economy started to recover from deep recession and it may take some time this year before the fragile pick-up in economic activity has an impact on reversing the trend in the labour market.

The baseline outputs: *however, it took nine months for the economy to get out of deep recession and it can take some time this year to have an impact on reversing trends in the labour market.* Dynamic fusion outputs: *but it has taken nine months for the economy to begin to turn out from the deep recession and this year it can take some time before suspending the trend in the labour market.* We can see here, that Dynamic Fusion is closer to the reference sentence and even though the last part of the sentence differs from the baseline and the reference, it still has a similar meaning.

Cold Fusion Variant 1 vs. Cold Fusion As can be seen in the results table, variant 1 does yield lower results than the vanilla Cold Fusion when using the Europarl language model. When comparing the outputs of our second run, we can see that in general both models tend to make similar mistakes, but sometimes different kinds of mistakes can also occur. Considering this vanilla Cold Fusion output:

> they have become second-class citizens who cannot speak in the public and afghanistan was once again a subject that was made in public.

Variant 1 produces: *they have become class people who can no longer be able to speak in the public and afghanistan was once again a member state and afghanistan was once a referendum.* It is evident, that variant 1 misses the 'second-class' aspect

and mentions afghanistan two times, while the second repetition makes no sense.
Variant 1 also produces different phrasings. Cold Fusion outputs:

> my question is therefore as follows. what do you intend to do to address all the irregularities and, above all, of course, to give you an end to it?

Variant 1 produces: *my question is this: what do you intend to do in order to deal with all of the irregularities. i would of course like to put an end to all of the irregularities.* It can be seen that variant 1 produces two sentences, the latter of which has a different meaning than the original sentence.

Gigaword LM vs. Europarl LM Models using the Gigaword language model suffer from repetitions. For example, in Cold Fusion we observed repetitions of three particular words:'84', 'april' and 'maystadt'. This is one of the main reasons for the lower BLEU score. We think this is related to the higher perplexity of the Gigaword language model. However, despite the lower results, the positive influence of the language model can be seen. Consider the reference translation:

> these countries have the right to choose the rate at which their borders are opened up and their economies liberalised.

The baseline produces: *these countries have a right to determine the speed at which their borders are opened and to the liberalisation of their economies.* Cold Fusion Variant 1 outputs: *these countries have a right to determine the pace of opening their borders and the liberalisation of their economy itself.* We can see that, instead of 'rate' or 'speed', cold fusion uses 'pace'. Cold Fusion also uses a different sentence structure in the second half of the sentence. The Cold Fusion output when using the Gigaword language model(*The commissioner has not mentioned 84 communication, which now appears 84 and which is not the subject of the international high level round held maystadt.*) shows the issue of frequent repetitions mentioned earlier. When comparing the results, we can see that the Europarl language model is clearly the better choice, since it yields higher results and the generated outputs are more readable.

5 Conclusion

Despite the lower results, one can clearly see the potential of models which combine translation and language models. Fusion methods can improve the fluency and the linguistic richness of the model's output.

However, we have also seen that one needs a highly trained language model for a successful combination with a translation model. Training the fusion models using two different language models highlights the importance of a good language model and the importance of training data. The output shows how a language model can support a translation model, but the linguistic analysis has also shown that there are negative effects like repetitions or useless insertions.

References

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *arXiv preprint arXiv:1503.03535*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proc. ACL*.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86. Citeseer.

Michiki Kurosawa and Mamoru Komachi. 2019. Dynamic fusion: Attentional language model for neural machine translation. *arXiv preprint arXiv:1909.04879*.

Anuroop Sriram, Heewoo Jun, Sanjeev Satheesh, and Adam Coates. 2017. Cold fusion: Training seq2seq models together with language models. *arXiv preprint arXiv:1708.06426*.

Shubham Toshniwal, Anjuli Kannan, Chung-Cheng Chiu, Yonghui Wu, Tara N Sainath, and Karen Livescu. 2018. A comparison of techniques for language model integration in encoder-decoder speech recognition. In *2018 IEEE Spoken Language Technology Workshop (SLT)*, pages 369–375. IEEE.

6 Appendix

The appendix provides the validation accuracy/perplexity plots of the models which used the Gigaword language model.

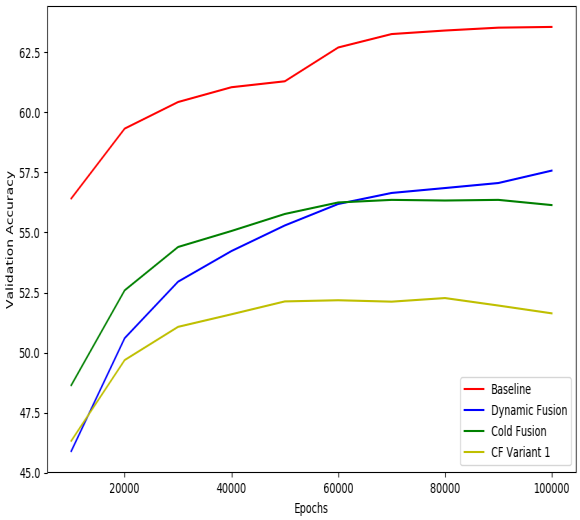

Figure 5: Validation accuracy Gigaword LM

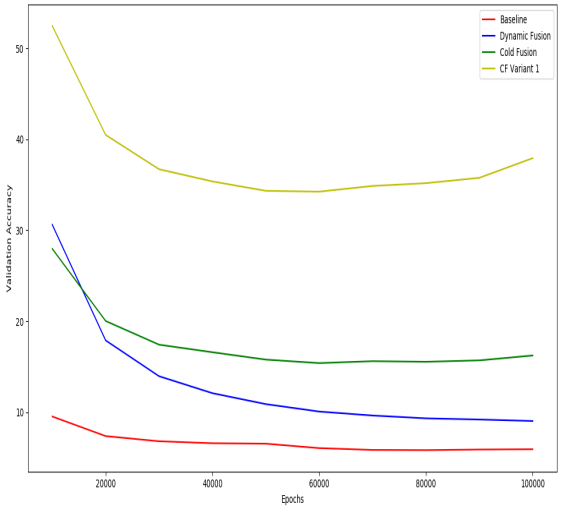

Figure 6: Validation Perplexity Gigaword LM