An Unsupervised Pretraining Task for the BiDAF Model

Stanford CS224N Default Project

Julius Stener
Masters Candidate in Computer Science
Stanford University
stenerj@stanford.edu

Abstract

The primary goal of this project is to determine whether pretraining a BiDAF model can improve its performance [1]. Over the past few years and particularly since "Attention is All You Need" was published, the NLP community has moved away from LSTM-based architectures because of the benefits seen by attention-only networks with extensive pre-training [2, 3]. The aim of this project was to determine if results can be improved on a BiDAF model simply by pretraining on a similar task to that used in the original BERT paper [4]. While the BERT paper used a Masked Language Model (MLM) and Next Sentence Predictions (NSP), this paper utilizes what I believe to be a novel variant of MLM, termed Obscured Replacement Language Model (ORLM), to achieve minor performance gains over baseline BiDAF training, as judged by the EM and F1 scores. Further, pretraining the BiDAF model with this method decreases the amount of training required on the SQuAD 2.0 training dataset to achieve similar performances, while boosting task-specific metrics such as the AvNA score.

1 Introduction

The current state-of-the-art natural language models are derived primarily from the paper "Attention Is All You Need," published by Vaswani et al. in December of 2017, and at that time, the current state-of-the-art was heavily based on recurrent networks which tracked hidden states over time [2, 1]. The problem the authors found with existing model was that it failed to parallelize well, and therefore limited the ability to train on a large dataset quickly. The solution to their problem was to extract the attention layer from the best models at the time and develop an attention network consisting of multiple attention layers stacked on top of one another into a "transformer," thereby yielding the name "Attention Is All You Need" [2].

Since then, the inherently parallelizable nature of training a transformer has enabled large gains in performance by simply pretraining the model on a set of unsupervised tasks, often one or both of Masked Language Model (MLM) and Next Sentence Prediction (NSP) [4, 5]. Further, research such as that by Raffel et al., in the colloquially named "T5" paper, highlights the limits of transfer learning to increase the performance of a transformer model [3]. The team systematically compared and analyzed the benefits of pretraining on their C4 dataset to achieve a baseline of comparison between the myriad of improvements on the Transformer model that had been published since the initial paper in 2017 [3].

The motivation for this paper was to determine whether transfer learning could reasonably be applied to the BiDAF model with tasks similar to those applied to a transformer model [1]. Of course, this does not negate Vaswani et al.'s original grievance about the lack of parallelizability of LSTM models; however, establishing that an unsupervised task can be utilized successfully would support that the community has not explored the full-potential of LSTM-based models [2].

As discussed in related work, this is not the first attempt to pretrain an LSTM or even a BiDAF model, but from my research of related work, the application of a fully unsupervised task to pretrain a BiDAF model is novel and could potentially provide a basis of further research with extended pretraining.

2 Related Work

Transfer learning, as discussed above, involves the training of a model on an upstream, often unsupervised task, before fine-tuning the model on the down-stream task it will be evaluated on. While transfer learning was pioneered nearly 50 years ago by Stevo Bozinovksi and Ante Fulgosi, transfer learning has become incredibly popular since the advent of highly-parallelizable models such as the transformer model because pretraining these models is considerably cheaper in both time and resources than previous model forms [6, 7]. Further, this provides a vast corpus of research which discusses the benefits of transfer learning on NLP tasks as well as research into the efficacy of models by removing unique transfer learning approaches [7, 3]. As discussed in the introduction, the T5 paper by Raffel et al. illustrates the benefits of transfer learning in enabling the model to learn character and word representations prior to seeing any down-stream tasks [3].

Furthermore, the most applicable previous work to this paper is authored by Min, Seo and Hajishirzi, published in 2017, and similarly analyzes the effects of transfer learning on the BiDAF model itself. Of note, they define a set of a supervised datasets, derived in part or in whole from existing QA datasets (such as SQuAD, WikiQA, SemEval and SICK), which allows them to pretrain models without changing the structure of the BiDAF model between pretraining and fine-tuning [8]. Their results were positive in that they showed pretraining can significantly increase the accuracy of the BiDAF model (with a nearly 30% increase in accuracy over baseline in some cases); however, the "accuracy" defined in the paper is not explicitly defined as either the EM or F1 score. As such, this paper unfortunately cannot provide a direct comparison between the improvements of those pretraining datasets and the results described below [8]. Yet, the paper does serve as an important baseline for understanding that BiDAF is absolutely capable of achieving significant gains from transfer learning on related tasks which are not the explicit task it would eventually be evaluated on.

3 Approach

Although not critically important to the research focus of this paper, I have implemented character-level embeddings for the BiDAF model, and the implementation submitted includes the ability to toggle between the full BiDAF model and the partial BiDAF model (i.e. without character level embeddings using the arg "−−full_bidaf") [1]. To achieve this layer, the model utilizes the provided character vectors to initialize an embedding layer, which is followed by a convolutional neural network layer activated by the ReLU function. As is described in the BiDAF paper, the layer utilizes a dropout layer during training [1]. Of note, the addition of character-level embeddings does improve the output of the model over the baseline. Given that the BiDAF model is discussed extensively both in the BiDAF paper and the assignment specifications, this paper will not repeat the architecture but it can be found easily here [1].

More critical to the topics of this paper, the pretraining approach discussed here differs from that of Min et al. in that it develops a task which is completely unsupervised, whereas Min et al. use exclusively supervised datasets [8]. The distincition between the use of supervised vs. unsupervised datasets to pretrain the model is notable because of the amount of text which can be efficiently collected into a novel pretraining dataset. Unsupervised tasks, by their very nature, do not require human intervention for every training example and have therefore paved the way for transfer learning over massive amounts of data. It is not an exaggeration to say that the field of NLP would not be where it is today without the use of unsupervised pretraining datasets.

Similar to the goals of unsupervised transformer pretraining, the goal of this task was to increase the character and word comprehension of the BiDAF model prior to fine-tuning on the SQuAD dataset. Of note, most unsupervised pretraining is optimized for transformers because the architecture of the transformer model is inherently capable of producing an output which is not correlated in length linearly with the input. By contrast, the BiDAF is limited to output a set of linear values which is inherently both not compatible with the output of word emeddings and of fixed length equal to the input context length. In this case, BiDAF pretraining is left with two options: (1) modify the

existing architecture by adding some form of a new output layer which enables the prediction of word embeddings or (2) define an unsupervised task which works within the existing model. Given the goal of transfer learning is to train the entire network to work together, the first option is less than ideal as it would require any fine-tuning to be the sole source of weigths for the model's output layer.

The second option is then superior, but the challenge then becomes that the BiDAF model is limited in the scope of the input and the output. The form of any such dataset must consist of a context, question and answer span from within the context. The solution explored in this paper is a variation of the Masked Language Model utilized extensively in the original BERT paper [4]. For the BiDAF-specific task, instead of having the model output the masked text itself as it does with the BERT transformer, the model in this case is meant to find the location within the context that text passed as the "question" should be inserted so as to complete the context correctly. The task itself, called Obscured Replacement Language Modeling (ORLM) and described further below, is for the BiDAF model to learn where the replacement should occur, and thereby this task is capable of working within the existing structure of the BiDAF model while also forcing it to learn word and character representations.

4 Experiments

4.1 Data

For pretraining, the dataset for Obscured Replacement Language Modeling (ORLM) is generated by parsing paragraphs of Wikipedia text, which can be found at the Wikimedia Latest English Downloads page, here. In the case of this paper, the small dataset consists of 300,000 unique context, question and answer triples from "enwiki-latest-pages-articles1.xml-p1p41242.bz2", and the large dataset consists of 1,000,000 unique context, question and answer triples from "enwiki-latest-pages-articles.xml.bz2".

After downloading and cleaning the dataset (which is all handled entirely in setup.py), the dataset is iterated through and every paragraph is randomly chosen to either be answerable or not answerable with probability 0.5. If the paragraph is set to be answerable, the paragraph becomes the context and a random-length string is removed from that context. This removed string becomes the question, and the space within the context that the string previously occupied is populated by a new, random-length string, which is generated as a random sample of the words in the original context. An example of this process is seen below. Although on average the context is equivalent in length before and after this process, the context is not guaranteed to be so. Finally, the answer to this context-question pair is then set to be the span containing the filled text. In practice, this paper set the length of the removed text and the fill text from a uniform distribution between 3 and 15 words long. A minimum of 3 words was used because I believed at the time that it would be too difficult for the model to learn otherwise. A lower minimum should be explored in future work as this was not the case.

Paragraph (Answerable):

```
"Everything I Wanted" received mainly positive reviews from music critics. The
song was praised by Insider's Callie Ahlgrim, who called it a "thoughtful
dynamic" and the lyrics a "breathtaking portrait of their in-sync
collaborative skills".
```

Question:

```
called it a "thoughtful dynamic" and the lyrics
```

Context:

```
"Everything I Wanted" received mainly positive reviews from music critics. The
song was praised by Insider's Callie Ahlgrim, who positive in-sync dynamic
skills Wanted a "breathtaking portrait of their in-sync collaborative skills".
```

Answer

```
positive in-sync dynamic skills Wanted
```

Conversely, if the paragraph is not going to be answerable, the paragraph is similarly set as the context, but a new, random-length string which is also a random sample of the words in the original

context is set to the be the question. Of course, no answer is set for this training example. Further, the exact same procedure is followed for generating the pretraining dev set of this dataset, but with the explicit rule than no paragraphs used in the pretraining set are used in the pretraining dev set. The fundamental idea behind the ORLM dataset is that the model must effectively "understand" the question's meaning to generate a correct location to insert it into the context, thereby facilitating the learning of character and word associations prior to fine-tuning.

For that fine-tuning, this paper exclusively utilized the training set provided within setup.py, with no alterations. This training set is the training set from the Stanford Question Answering Dataset Version 2.0 (SQuAD 2.0) and is made up of of roughly 128,000 context and question pairs, of which roughly half are answerable [9]. As with the pretraining dataset, SQuAD 2.0 is drawn from the Wikipedia corpus, and therefore it can be reasonably assumed that it consists of well-formed text. However, unlike the pretraining dataset, SQuAD 2.0 could potentially repeat the same context with different queries, and additionally, SQuAD 2.0 provides multiple (4) answers to every answerable query because answers which diverge by one word in the QA task are not necessarily incorrect [9]. Quite deliberately, the reason that the pretraining dataset does not ever repeat contexts between training examples is that we want to minimize the opportunity for the network to learn the specific example (despite little evidence that LSTM models learn specific examples).

4.2 Evaluation method

For evaluation, the baseline models and the transfer learning models are compared utilizing Exact Match (EM), F1 and AvNA scores. As the name implies, Exact Match on a single training example is 1 when the model output matches the provided answer and 0 otherwise. On average, it indicates how often the model correctly identifies the exact span given by one of the possible answers, and it is therefore the strictest measure. For a slightly more forgiving measure, the F1 score provides a combination of the precision and recall of the output against the provided answer, as described in the default project overview. Given the forgiving nature of human responses to questions, the F1 score is the best for evaluating whether of not a model is learning to identify the correct spans within the context. Lastly, the Answer vs. No Answer score is the percentage of time the model correctly predicts the answer is or is not within the provided context. While less directly correlated to a model's understanding of the context-question pair, the AvNA score provides a useful metric for understanding the degree to which a model "guesses" at an answer vs. identifies an answer.

4.3 Experimental details

Given that the goal of this paper is to determine the extent to which transfer learning on an unsupervised task can be used to improve the performance of the BiDAF model, the majority of hyper-parameters between experiments did not vary, just as they don't in other papers seeking to compare models such as the T5 paper [3]. Below, "Partial-BiDAF" refers to the BiDAF model without the implementation of character embeddings, and the "Full-BiDAF" model naturally includes these embeddings. "Pretraining" is performed on the ORLM dataset and task described above, and fine-tuning is performed on the same training set as the Partial and Full-BiDAF baselines trained on. Given the lack of hyper-parameter and fine-tuning variation, the experiment results can all be compared to determine the effects of transfer learning on the BiDAF model.

Model	# Training Ex.	# Steps	LR	NLL	Batch	Hidden
Partial-BiDAF Training Baseline	128,000	3,890,000	0.5	3.16	64	100
Full-BiDAF Training Baseline	128,000	3,890,000	0.5	3.04	64	100
Full-BiDAF Pretraining - Small	300,000	3,000,000	0.5	2.16	64	100
Full-BiDAF Pretraining - Large	1,000,000	4,500,000	0.5	0.76	64	100
Full-BiDAF Fine-Tuning - Small	128,000	3,890,000	0.5	3.02	64	100
Full-BiDAF Fine-Tuning - Large	128,000	3,890,000	0.5	2.89	64	100

Of note, the "Full-BiDAF Pretraining - Large" training was cut short due to both a lack of credits as a well as a lack of time at the end of this project. Having only trained through the dataset 4.5 times, I do not feel that the results from the "Large" pretraining example is the best reflection of how well transfer learning can perform on a dataset of that size simply because the model was not given enough training to adequately learn that dataset.

4.4 Results

The results of the experiments can be seen in the table below. The best performing model, Full-BiDAF Pretrained Large, achieving an F1 score of 63.83 and a EM score of 59.205 on the test set submitted to the Gradescope Default Project Final Project - IID SQuAD Track test leaderboard under the name "JTS BiDAF Fine-Tuned-Large". On the dev set, this model also achieved an F1 score of 63.828 and a EM score of 60.628 also submitted under the name "JTS BiDAF Fine-Tuned-Large" at the Gradescope Default Project Final Project - IID SQuAD Track dev leaderboard. NLL in the table below is reported on the dev set.

Data Set	Model	F1	EM	AvNA	NLL
Dev Set	Partial-BiDAF Baseline	60.77	57.47	67.50	3.16
Dev Set	Full-BiDAF Baseline	62.82	59.54	68.95	3.04
Dev Set	Full-BiDAF Pretrained-Small	63.06	59.75	69.67	3.02
Dev Set	Full-BiDAF Pretrained-Large	**63.83**	**60.63**	**70.07**	**2.89**
Test Set	Full-BiDAF Pretrained-Large	**63.828**	59.205	N/A	N/A

Clearly from the results of the experiments, performing pretraining on the ORLM task marginally improves the F1, EM and AvNA scores that the BiDAF model can achieve. While the terms "Small" and "Large" are potentially misleading in that they imply the model itself has a fewer or greater number of parameters, these terms indicate the relative size and duration of pretraining dataset and pretraining time, respectively. As can be seen, increasing the size and the training steps the model is exposed to during pretraining does correlate with an increase in the metrics that SQuAD provides, and this would support that more pretraining over a longer period of time with a ORLM larger dataset would yield even better results. That being what it is, even pretraining on the 1,000,000 pretraining examples for just 4.5 passes through the dataset yields an increase in the F1 score by over 1%, which reflects nearly 50% of the increase that is gained by implementing character-level embeddings in the BiDAF model. Similarly, pretraining on the ORLM task and large dataset yields an increase in the AvNA score of 1.12% which accounts for nearly 80% of the increase had by adding character-level embeddings to the Partial-BiDAF baseline.

Furthermore, the fact that the F1 scores for the both dev and test set are the exact same indicates that the model was not over-trained to the dev set. Notably though, I do not know why the Full-BiDAF Pretrained Large performed so much worse on the test set (59.205) than on the dev set (60.628) for the Exact Match score. The current theory is that the pretraining biased the model to output sequences which are ill-formed because that is what the ORLM task necessitated. This is a very real potential negative to the ORLM as an unsupervised pretraining task. As such, the model would likely continue to seek out ill-formed sequences in examples it has not seen before on the test; yet, even this does not explain why the model would perform so well on the dev set at the same time.

As expected, the model demonstrated that it is capable of learning character and word associations prior to fine-tuning; however although the model did perform better, the degree to which the pretraining increased the scores leaves some room for further improvement to be desired. While the approach is sound, further pretraining on even larger datasets is likely to yield better results. Furthermore, as can be seen in the figure below, the scores are in fact consistently better than the scores of models without pretraining.

5 Analysis

For the qualitative analysis, considerable literature exists for understanding benefits of adding character-level embeddings to a model, particularly as it is applied to the BiDAF model [1]. Therefore, this section will focus primarily on identifying the qualitative differences between the Full-BiDAF-Baseline implementation (i.e. with character-level embeddings but without pretraining) and the Full-BiDAF-Fine-Tuned-Large implementation (i.e. with character-level embeddings and with pretraining).

On the vast majority (>90%) of context-question pairs in the dev set, the pretrained and the baseline BiDAF models output the exact same answer. This is exactly what would be expected to occur between models with the exact same architecture and is nothing to be concerned/curious about. However, for the last 10% of examples in the dev set that do differ, it is crucial to explore why they

differ. Within the appendix there are 6 full-examples in which the context-question-answer triples where the Full-BiDAF baseline and the Full-BiDAF fine tuned large differ in predictions. Of note, there are multiple examples in which one of the baseline or the fine-tuned model are incorrect and the other is correct; however, when the fine-tuned model is correct, it is more likely to match closer to the gold-medal answer. Specifically in reference to examples number 2 and number 3, it is plain that the understanding of the language is ever-so-slightly better in cases in which modifiers of the technical answer play an important role in understanding the answer. For instance in the second example, while "1970s" is in fact correct, the fine-tuned output of the "By the 1970s" is far more applicable an answer to the query "When had?"

Interestingly, across all of the dev set, the fine-tuned model is considerably less likely to identify single word answers than the baseline model is. This is likely a carry-over fault from when I established the minimum number of words in the questions and answers in the ORLM dataset to be 3 words. As a result, the model penalizes answers which are too short, often leading to incorrect predictions simply because the fine-tuned model is searching for a longer length string. Although time and resources didn't permit the exploration of this during this project, an interesting future work would be to run the pretraining again, setting the minimum number of words in the answer to be 1 or 0 as well as biasing the number of words in the answer of the ORLM dataset to be lower than the existing 9 words. Both of these changes would map the ORLM dataset more closely to the SQuAD 2.0 dataset and would likely improve the performance of a model pretrained by this task.

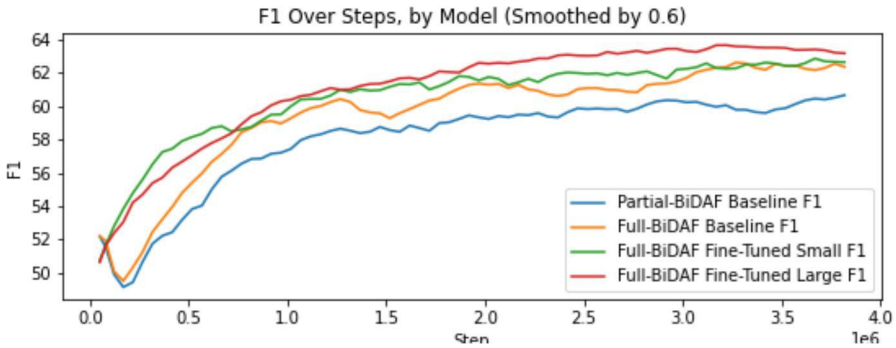

Figure 1: F1 Over Steps, by Model and Smoother by 0.6

Note: For further plots of each of these scores on the dev set over the steps of training, see the Appendix.

Finally, in the analysis of the training, it must be noted that pretraining the model increases the stability of the fine-tuning on the training set. Without pretraining, the model struggles initially across all metrics because it starts to learn to simply predict No Answer on all dev examples. The pretraining ORLM task takes care of this issue because the model learns to identify the difference between Answerable (well-formed relative to the context) and Not Answerable (non-well formed relative to the context) questions before it ever trains on the fine tuning dataset. The final benefit to this is that a pretrained model can be fine-tuned for fewer iterations than a model that wasn't pretrained before the either achieves its optimum score.

6 Conclusion

In short, this paper presents a novel, unsupervised task which can be used to effectively pretrain a standard BiDAF model and improve its EM, F1 and AvNA scores. The primary goal of this paper was to establish an unsupervised pretraining task, in this case ORLM, to determine if the BiDAF model could be improved through transfer learning. While the increases in performance from a modest amount of pretraining are not drastic, the paper provides evidence that the full potential of the BiDAF model to effectively perform the QA task has not yet been explored because there simply have not been enough attempts at large-scale pretraining of the model. Furthermore, this is contextually relevant to the current state of the art because current transformer architectures are aided massively by the extent to which they are pretrained, and theoretically, if it we possible to parrellize the BiDAF

or similar LSTM-based architectures and pretrain them to the same extent, transformer models may not be as stand out as they are today.

That said, the primary limitations of this work are the lack of time spent pretraining and of course the limitations of training the architecture itself. Compared with modern state-of-the-art transformer architectures, the BiDAF model consists of orders-of-magnitude fewer weights by which to learn the word and character representations, and although it has been scaled by researchers since the original publishing of the paper, the lack of parallelizable training continues to keep the model sizes small relative to something like BERT [1, 4].

Lastly and although throughout this paper there have hints towards avenues for future work, the best options for future work are: (1) evaluate the effectiveness of the ORLM task when the answer length minimum is set to 0 or 1, (2) explore pretraining on large, less-well-formed ORLM style datasets such as an adapted C4 dataset, and (3) explore how increasing the hidden size as well as the number of attention layers affects the models ability to learn more from pretraining.

References

[1] Ali Farhadi Hannaneh Hajishirzi Minjoon Seo, Aniruddha Kembhavi. Bidirectional attention flow for machine learning. In *arXiv*, 2016.

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

[3] Adam Roberts Katherine Lee Sharan Narang Michael Matena Yanqi Zhou Wei Li Peter J. Liu Colin Raffel, Noam Shazeer. Exploring the limits of transfer learning with a unified text-to-text transformer. In *arXiv*, 2020.

[4] Kenton Lee Kristina Toutanova Jacob Devlin, Ming-Wei Chang. Bert: Pre-training of deep bidirectional transformers for language understanding. In *arXiv*, 2018.

[5] Patrick Xia Raghavendra Pappagari R. Thomas McCoy Roma Patel Najoung Kim Ian Tenney Yinghui Huang Katherin Yu Shuning Jin Berlin Chen Benjamin Van Durme Edouard Grave Ellie Pavlick Samuel R. Bowman Alex Wang, Jan Hula. Can you tell me how to get past sesame street? sentence-level pretraining beyond language modeling. In *arXiv*, 2018.

[6] Ante Fulgosi Stevo. Bozinovski. The influence of pattern similarity and transfer learning upon the training of a base perceptron b2. In *Proceedings of Symposium Informatica*, 1976.

[7] Keyu Duan Dongbo Xi Yongchun Zhu Hengshu Zhu Hui Xiong Qing He Fuzhen Zhuang, Zhiyuan Qi. A comprehensive survey on transfer learning. In *arXiv*, 2019.

[8] Hannaneh Hajishirzi Sewon Min, Minjoon Seo. Question answering through transfer learning from large fine-grained supervision data. In *arXiv*, 2017.

[9] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Association for Computational Linguistics (ACL)*, 2018.

A Appendix (optional)

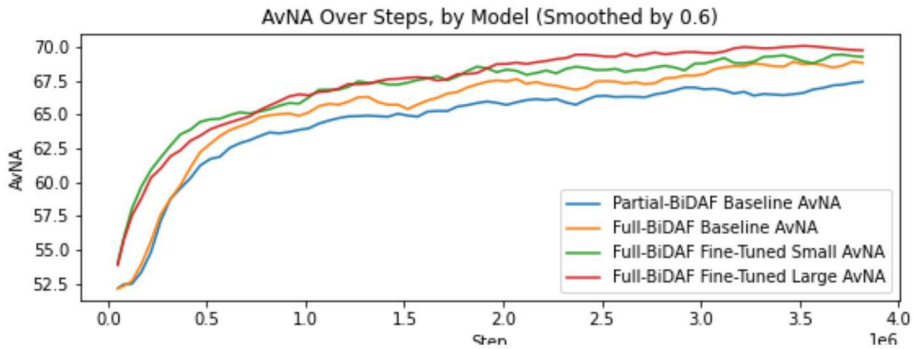

Figure 2: AvNA Over Steps, by Model and Smoother by 0.6

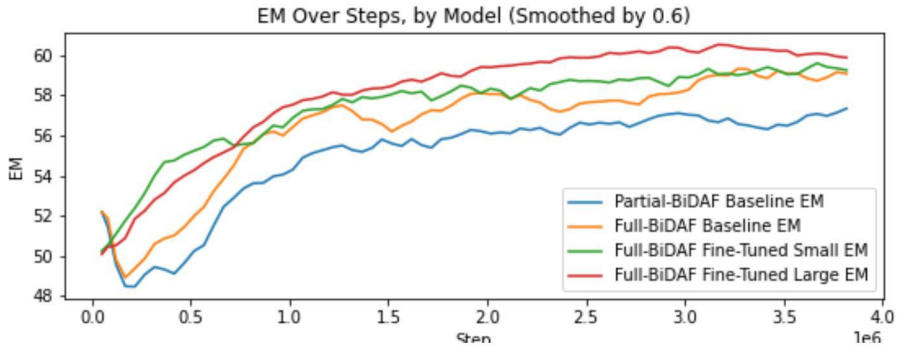

Figure 3: EM Over Steps, by Model and Smoother by 0.6

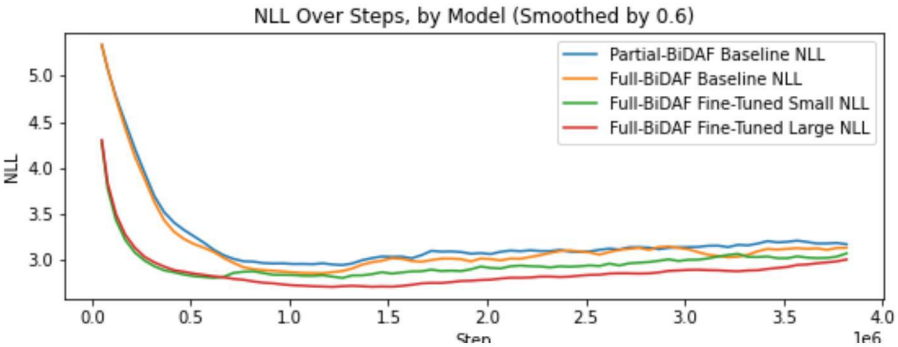

Figure 4: NLL Over Steps, by Model and Smoother by 0.6

EXAMPLE 1

Example Question:

`Economy, Energy and Tourism is one of the what?`

Example Context:

`Subject Committees are established at the beginning of each parliamentary session, and again the members on each committee reflect the balance of parties across Parliament. Typically each committee corresponds with one (or more) of the departments (or ministries) of the Scottish Government. The current Subject Committees in the fourth Session are: Economy, Energy and Tourism; Education and Culture; Health and Sport; Justice; Local Government and Regeneration; Rural Affairs, Climate Change and Environment; Welfare Reform; and Infrastructure and`

Capital Investment.

Correct Answer:

current Subject Committees

Full-BiDAF Baseline Prediction:

Subject Committees in the fourth Session

Full-BiDAF Fine Tuned Large Prediction:

N/A

EXAMPLE 2

Example Question:

When had the Brotherhood renounced violence as a means of achieving its goals?

Example Context:

While Qutb's ideas became increasingly radical during his imprisonment prior to his execution in 1966, the leadership of the Brotherhood, led by Hasan al-Hudaybi, remained moderate and interested in political negotiation and activism. Fringe or splinter movements inspired by the final writings of Qutb in the mid-1960s (particularly the manifesto Milestones, a.k.a. Ma'alim fi-l-Tariq) did, however, develop and they pursued a more radical direction. By the 1970s, the Brotherhood had renounced violence as a means of achieving its goals.

Correct Answer:

By the 1970s

Full-BiDAF Baseline Prediction:

1970s

Full-BiDAF Fine Tuned Large Prediction:

By the 1970s

EXAMPLE 3

Example Question:

When did Germany invade Poland and in doing so start World War II?

Example Context:

After the German Invasion of Poland on 1 September 1939 began the Second World War, Warsaw was defended till September 27. Central Poland, including Warsaw, came under the rule of the General Government, a German Nazi colonial administration. All higher education institutions were immediately closed and Warsaw's entire Jewish population - several hundred thousand, some 30\% of the city - herded into the Warsaw Ghetto. The city would become the centre of urban resistance to Nazi rule in occupied Europe. When the order came to annihilate the ghetto as part of Hitler's "Final Solution" on 19 April 1943, Jewish fighters launched the Warsaw Ghetto Uprising. Despite being heavily outgunned and outnumbered, the Ghetto held out for almost a month. When the fighting ended, almost all survivors were massacred, with only a few managing to escape or hide.

Correct Answer:

September 1939

Full-BiDAF Baseline Prediction:

N/A

Full-BiDAF Fine Tuned Large Prediction:

1 September 1939

EXAMPLE 4

Example Question:

How did peace start?

Example Context:

The war was fought primarily along the frontiers between New France and the British colonies, from Virginia in the South to Nova Scotia in the North. It began with a dispute over control of the confluence of the Allegheny and Monongahela rivers, called the Forks of the Ohio, and the site of the French Fort Duquesne and present-day Pittsburgh, Pennsylvania. The dispute erupted into violence in the Battle of Jumonville Glen in May 1754, during which Virginia militiamen under the command of 22-year-old George Washington ambushed a French patrol.

Correct Answer:

N/A

Full-BiDAF Baseline Prediction:

violence

Full-BiDAF Fine Tuned Large Prediction:

over control of the confluence of the Allegheny and Monongahela rivers

EXAMPLE 5

Example Question:

What is the Norman architecture idiom?

Example Context:

Norman architecture typically stands out as a new stage in the architectural history of the regions they subdued. They spread a unique Romanesque idiom to England and Italy, and the encastellation of these regions with keeps in their north French style fundamentally altered the military landscape. Their style was characterised by rounded arches, particularly over windows and doorways, and massive proportions.

Correct Answer:

Romanesque

Full-BiDAF Baseline Prediction:

Romanesque

Full-BiDAF Fine Tuned Large Prediction:

a new stage in the architectural history of the regions they subdued

EXAMPLE 6

Example Question:

Who proposed that water displaced through the projectile's path carries the projectile to its target?

Example Context:

Aristotle provided a philosophical discussion of the concept of a force as an integral part of Aristotelian cosmology. In Aristotle's view, the terrestrial sphere contained four elements that come to rest at different "natural places" therein. Aristotle believed that motionless objects on Earth, those composed mostly of the elements earth and water, to be in their natural place on the ground and that they will stay that way if left alone. He distinguished between the innate tendency of objects to find their "natural place" (e.g., for heavy bodies to fall), which led to "natural motion", and unnatural or forced motion, which required continued application of a force. This theory, based on the everyday experience of how objects move, such as the constant application of a force needed to keep a cart moving, had conceptual trouble accounting for the behavior of projectiles, such as the flight of arrows. The place where the archer moves the projectile was at the start of the flight, and while the projectile sailed through the air, no discernible efficient cause acts on it. Aristotle was aware of this problem and proposed that the air displaced through the projectile's path carries the projectile to its target. This explanation demands a continuum like air for change of place in general.

Correct Answer:

N/A

Full-BiDAF Baseline Prediction:

Aristotle

Full-BiDAF Fine Tuned Large Prediction:

N/A