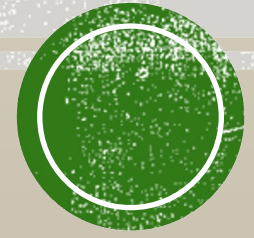
Question Answering System Based on HOTPOTQA



Jiaming Li, student ID 300233503

CSI 5180 - Topics in AI: Virtual Assistant

Master of Computer Science, University of Ottawa

Final Project Presentation

Content

- Project Summary
- Methodology
- Activity Table
- Challenges and Resolutions
- Demo, Results and Outputs
- Conclusion

1. Project Summary

- Question answering (QA) is a significant Natural language processing (NLP) problem as well as a long-standing AI milestone.
- Reading comprehension is challenging for robots because it necessitates a combination of natural language comprehension and world knowledge.
- The project modeled HotpotQA dataset as a seq2seq problem to learn a piece of text, and artificially asks some related questions based on the content of the article.
- The model provides answers to these relevant questions based on the content of the learned articles.

2. Methodology

- HotpotQA Dataset
  - Context
  - Question
  - Supporting facts
  - Answer
- Convolutional Seq2Seq Networks
  - Positional Embeddings
  - Residual connections

- Project Settings
  - Software Platform
    - PyCharm, macOS, Github
  - Programming Environment
    - Python 3.9.0
    - PyTorch 1.10.2
    - pytorch-transformers 1.2.0
  - Datasets - HotpotQA

- Parameters Settings

Settings	Baseline	With Attention
Batch Size	128	32
Hidden Dim	100	50
Encoder Embedded Dim	100	50
Decoder Embedded Dim	100	50
Encoder Dropout	0.5	0.5
Decoder Dropout	0.5	0.5
Epochs	10	10
Clip	1	1

Table 1: Parameters Settings for the Models

3. Activity table

Activity	Why	Time Taken	Deliverable
Research and study current articles and achievements, build background knowledge	Gather knowledge about QA system models	5 hours	None (Online resources and free video tutorial)
Explore and compare different datasets	Decide which dataset is the most suitable to the project	2 hours	Chose HOTPOTQA
Install the libraries and packages that used in this project and configure the environment	QA system requires a proper environment to run the program	2 hours	The projects can worked well on my laptop
Build and run the model (baseline)	Turn the QA process into a function so that the examples can be easily tried	6 hours	Some result plots should be generated
Build and run the model (with attention)	Improve the model and compare the results with baseline	4 hours	Some result plots should be generated
Test some examples of the model and compare the result both technically and artificially	See how accurate the model could be	3 hours	Answers of questions that I give to the model
Fix and improve the model	Get better results	2 hours	Output could be more accurate
Write report and record the presentation for final project	Summarize what I do for this project	7 hours	Video and paper
Total hours: 31 hours			

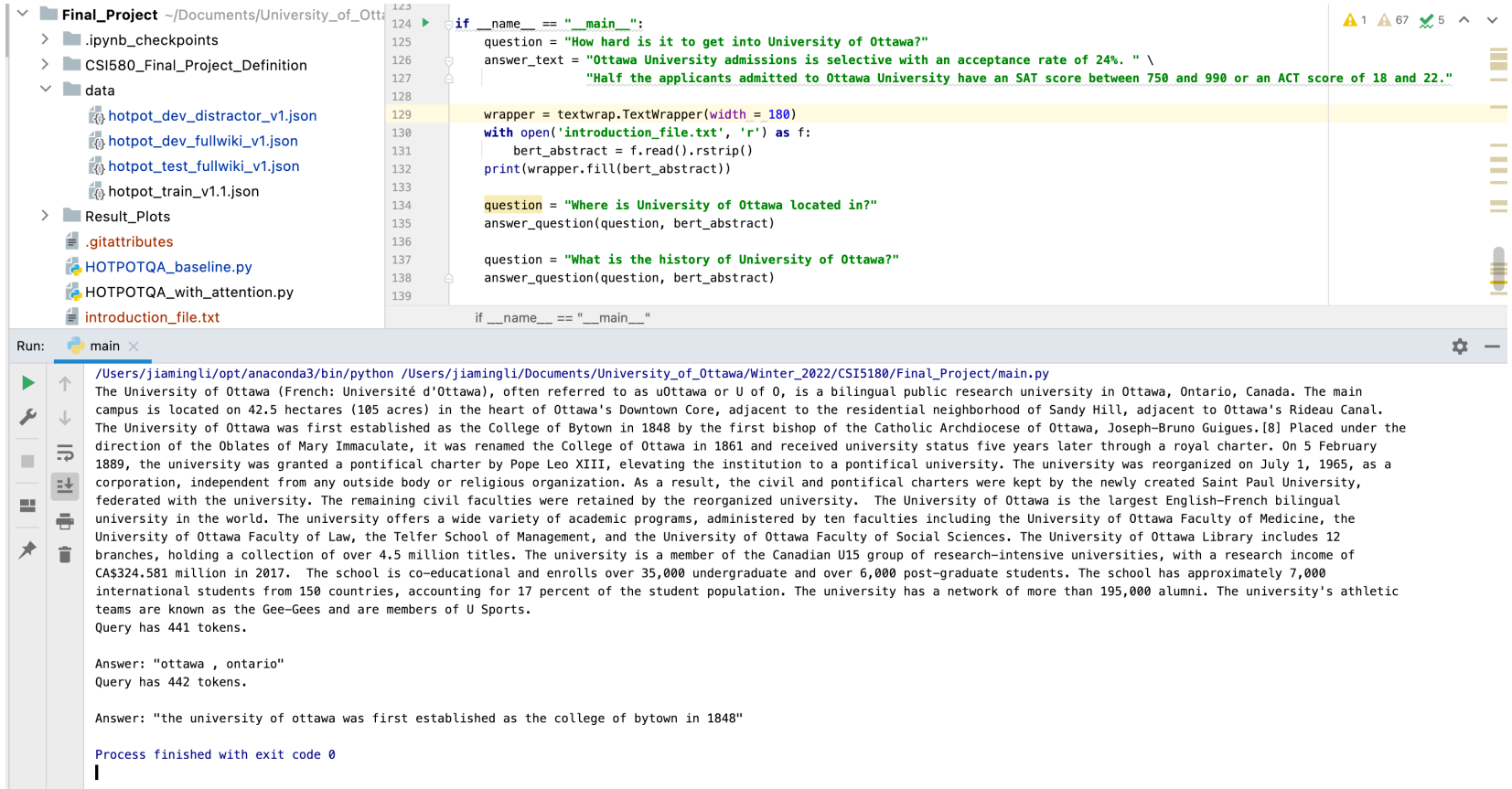
Table 2: Activity Table

4. Challenges and Resolutions

- (Resolutions: What I have learned)
- Dataset Changing
  - Many prominent datasets, such as SQuAD [1], have been created with the goal of providing a dataset to train reading comprehension models.
  - To get around these restrictions, HotpotQA creates questions that involve reasoning across numerous sources in natural language, rather than relying on a pre-existing knowledge base [2].
- Researches on HotpotQA are limited
  - The number of papers on HotpotQA is very limited, I did some research on building a model based on HotpotQA dataset.
    - Simple and effective multi-paragraph reading comprehension [3]
    - Attention is all you need [4]

5. Demo, Results and Outputs

- 5.1 Demo
  - Figure 1 shows what I tested for the model.



5. Demo, Results and Output

- 5.2 Results
  - The following tables show the results of loss and perplexity values.

Epochs	Baseline Train Loss	Valid Loss	with Attention Train Loss	Valid Loss
1	5.834	5.733	5.737	5.732
2	5.130	5.653	5.142	5.632
3	4.836	5.541	4.841	5.533
4	4.560	5.435	4.565	5.461
5	4.338	5.418	4.359	5.489
6	4.145	5.432	4.195	5.521
7	3.963	5.453	4.056	5.507
8	3.795	5.493	3.926	5.542
9	3.595	5.582	3.816	5.589
10	3.401	5.687	3.693	5.597

Table 3: Comparison of Train and Valid Loss of Two Models

Epochs	Baseline Train PPL	Valid PPL	with Attention Train PPL	Valid PPL
1	341.765	308.765	310.068	308.443
2	168.942	285.170	171.088	279.133
3	125.912	254.845	126.599	252.864
4	95.623	229.224	96.104	235.350
5	76.550	225.499	78.191	242.006
6	63.122	228.493	66.321	249.995
7	52.631	233.348	57.764	246.381
8	44.495	242.908	50.704	255.139
9	36.429	265.648	45.424	267.489
10	29.986	295.011	40.158	269.701

Table 4: Comparison of Train and Valid Perplexity Values of Two Models

5. Demo, Results and Output

- 5.3 Outputs
  - The plots of train and valid loss with/without attention, train and valid perplexity values with/without attention are displayed below:

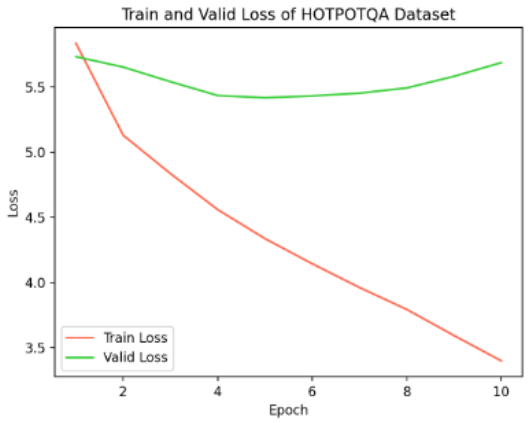


Figure 2: Train and Valid Loss

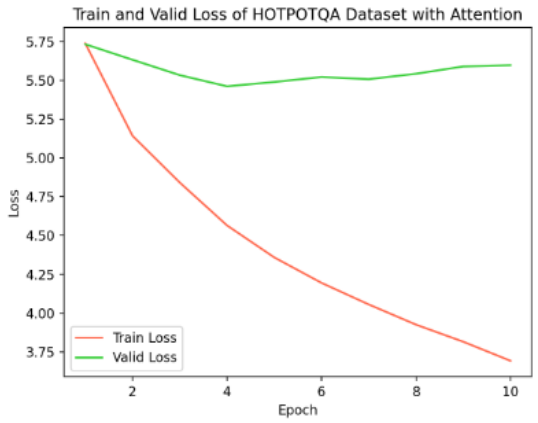


Figure 3: Train and Valid Loss with Attention

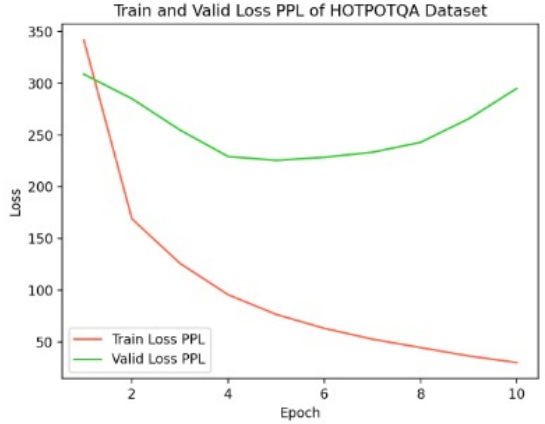


Figure 4: Train and Valid Perplexity Values

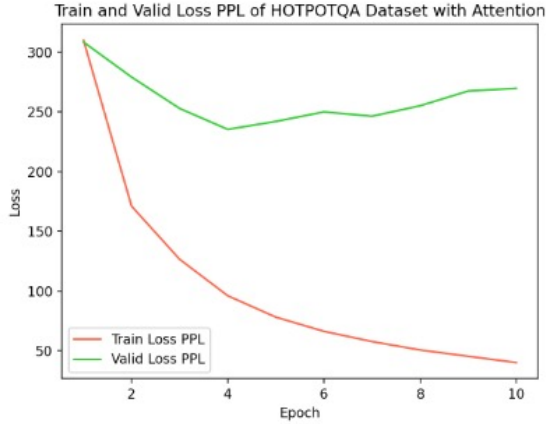


Figure 5: Train and Valid Perplexity Values with Attention

6. Conclusion

- Identified some issues with the baseline model based on the HotpotQA dataset
- Proposed and implemented some learning and architectural changes to that model
- With attention, loss value and perplexity value are higher than the values of baseline, which does not change efficiently

Reference

- [1] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," in Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2016, pp. 2383–2392. [Online]. Available: http://aclweb.org/anthology/D16-1264
- [2] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. W. Cohen, R. Salakhutdinov, and C. D. Manning, "Hotpotqa: A dataset for diverse, explainable multi-hop question answering," 2018.
- [3] Clark, C., & Gardner, M. (2017). Simple and effective multi-paragraph reading comprehension. arXiv preprint arXiv:1710.10723.
- [4] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.
- [5] Rajpurkar, P., Jia, R., & Liang, P. (2018). Know what you don't know: Unanswerable questions for SQuAD. arXiv preprint arXiv:1806.03822.
- [6] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Rush, A. M. (2019). Huggingface's transformers: State-of-the-art natural language processing. arXiv preprint arXiv:1910.03771.
- [7] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Rush, A. M. (2020, October). Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations (pp. 38-45).
- [8] Su, D., Xu, Y., Winata, G. I., Xu, P., Kim, H., Liu, Z., & Fung, P. (2019, November). Generalizing question answering system with pre-trained language model fine-tuning. In Proceedings of the 2nd Workshop on Machine Reading for Question Answering (pp. 203-211).
- **My GitHub Code: https://github.com/JaneLi99/CSI5180_Final_Project**

THANKS

Any comments and suggestions are welcome.