Table Cell Classification for Question Answering

1 Motivation

The motivation for this assignment is to give you practice with developing and implementing new neural models for a given input/output structure. We do this for the task of tabular QA.

The ability to understand and process tabular data is crucial in many fields, including finance, healthcare, and science. With the vast amount of information stored in tables, there's a growing need for models that can accurately interpret and answer questions based on tabular data. Further, tables are 2 dimensional and direct use of sequence to sequence language models may not be the optimal approach. This assignment aims to give hands-on experience with this challenge, focusing on classifying table cells that contribute to answering a given question.

2 Problem Statement

The task is related to Table question answering. In particular each question is accompanied by a database table, containing certain information. Given the question, you have to output the correct answer. However, to make things simpler instead of answering the correct answer, you have to classify which cells correspond to the right answer.

You are tasked with developing a classification model that, given a question and a corresponding database table, identifies which cells correspond to the correct answer. For example, given a database table on cricket, and question: "Which teams defeated their opponents by more than 50 runs?" Your model should classify each cell in the table if it answers the question posed. That is, you will be classifying all cells of teams which meet the condition as positive while other cells as negative. You need to use deep learning for this assignment. The allowed models are CNN, LSTM, Transformers (or any extension of these, including attention or custom-built models). However, you are **not** allowed to use any pretrained language models.

2.1 Characteristics of the Task:

- Each question is associated with a unique table, which may vary in schema and size. This means, you cannot use a fixed classification layer, but require to encode both the label space and input space, and look for similarity in between them. For training you may want to use negative sampling.
- The table is presented as a dictionary, requiring preprocessing to extract relevant information.
- For each <table, question> pair, multiple cells may be correct making it a multi-label problem.

- The output should classify relevant rows, columns, and cells, with the final accuracy being the mean of these individual accuracies. Note, we will use subset accuracy for evaluation.
- The table may contain a mix of different types of data such as numerical, date, textual, etc. You are advised to handle this heterogeneity, in your solution.
- A single table may be linked to multiple questions. Additionally, metadata is provided for each table, which you can utilize.

3 Data

We are providing a train dataset with 25K questions and a val set of 5K questions. Each data point is accompanied with corresponding tables, and gold answers. Figure 1 shows representative example from the dataset.

	Player	Team	Matches	Overs	Wickets	Average	Economy	BBI	4wi	5wi
0	Virender Sehwag Category:Articles with hCards	Leicestershire	6	14.0	7	10.00	5.00	3/14	0	0
1	Jimmy Ormond Category:Articles with hCards	Surrey	5	20.0	11	10.09	5.55	5/26	1	1
2	Azhar Mahmood Category:Articles with hCards	Surrey	5	18.5	12	10.25	6.53	4/20	1	0
3	Dominic Hewson Category:Articles with hCards	Derbyshire	5	19.0	10	10.90	5.73	4/18	1	0
4	Jason Brown Category:Articles with hCards	Northamptonshire	5	17.5	11	11.09	6.84	5/27	0	1
5	Glen Chapple Category:Articles with hCards	Lancashire	5	19.0	9	11.22	5.31	2/13	0	0
6	Ashley Noffke Category:Articles with hCards	Middlesex	3	12.0	8	12.12	8.08	3/22	0	0
7	Peter Martin Category:Articles with hCards	Lancashire	4	15.0	7	12.14	5.66	3/20	0	0
8	Adam Hollioake Category:Articles with hCards	Surrey	7	25.1	16	12.31	7.82	5/21	1	1
9	Simon Cook Category:Articles with hCards	Middlesex	5	18.2	9	13.77	6.76	3/14	0	0

Figure 1: Example Table. Question: What are the lowest number of wickets? Correct labels are marked in green.

	Average population (x 1000)	Live births	Deaths	Natural change	Crude birth rate (per 1000)	Crude death rate (per 1000)	Natural change (per 1000)
0	233	6,559	1,938	4,621	28.2	8.3	19.8
1	253	6,950	2,306	4,644	27.5	9.1	18.4
2	272	7,133	2,748	4,385	26.2	10.1	16.1
3	287	8,110	2,624	5,486	28.3	9.1	19.1
4	309	8,116	2,664	5,452	26.3	8.6	17.7
5	304	7,271	2,873	4,398	23.9	9.5	14.5
6	303	6,545	3,006	3,539	21.6	9.9	11.7
7	302	6,130	3,480	2,650	20.3	11.5	8.8
8	303	6,076	4,086	1,990	20.1	13.5	6.6
9	304	6,172	4,010	2,162	20.3	13.2	7.1

Figure 2: Example Table. Question: What is sum of natural change in years when natural change per 1000 was less than 14.

4 Implementation Details

4.1 Allowed Libraries:

- PyTorch for model building
- Numpy for numerical operations
- gensim for accessing Glove, word2vec, or FastText embeddings
- pandas for handling tabular data

- Any other standard Python library included in the default installation

Any additional packages can be requested over Piazza and are only allowed **after verification**.

4.2 Output Format:

Your model's predictions should be saved in a JSONL file, with each line formatted as follows:

```
{ "qid": "<question_id>", "label_row": ["<correct_rows>"], "label_col": ["<correct_cols>"],
"label_cell": [["<row>", "<col>"]] }
```

A sample prediction file is provided for reference.

5 Evaluation Criteria

Your model will be evaluated based on the subset (exact match) accuracy of row, column, and cell predictions individually, with the final score being the mean of these three accuracies. All evaluations will follow an all-or-nothing approach, rewarding only completely correct classifications for each question.

6 Submission Instructions

6.1 Deadline

The assignment must be submitted by [18th March, 2024], 11:59 PM.

6.2 Submission Format

Submit a zip file named <kerberos_id>.zip through Moodle. Upon unzipping, it should create a directory named <kerberos_id>containing the following files:

- `install_requirements.sh` for setting up the environment
- `run_model.sh` for training and testing the model
- `writeup.txt` detailing your approach, discussions, citations, references, libraries used, and all hyperparameters used.

6.3 Training and Inference Commands

- To train the model, use: `bash run_model.sh <train_file> <val_file>`. This should save your model and any necessary files for inference.
- For inference, use: `bash run_model.sh test <test_file> <pred_file>`, which should produce the prediction file in the specified output format.

7 Submission Guidelines

Your submission must adhere to the following instructions:

- The assignment is to be done individually.

- You should use **Python 3.10** for this assignment. Only aforementioned packages are allowed to be used. Any additional packages can be requested over Piazza and are only allowed after verification.
- Using any external source of data is prohibited.
- Use of language models or other pretrained generative models for generating synthetic data is **strictly prohibited.**
- Your code will be tested on a Linux system with 16GB of memory, 4 cores and 8GB of GPU memory. You are responsible for making sure that it runs smoothly on Linux without any errors.
- You must not discuss this assignment with anyone outside the class. Make sure you mention the names in your write-up in case you discuss with anyone from within the class. Please read academic integrity guidelines on the course home page and follow them carefully.
- We will run plagiarism detection software. Any person found guilty will be awarded a suitable penalty as per IIT rules.
- Your code will be automatically evaluated. You will get a 20% flat penalty if it does not conform to output guidelines.
- Your code must be your own. You are not allowed to use ChatGPT or any coding assistant or large language models for this assignment. We will run plagiarism detection on the model generated code, and any similarity will be penalized as per IIT rules.

8 Clarifications

For any doubts or clarifications, please use the Piazza forum. Ensure your code is well-commented, follows the submission guidelines, and is tested thoroughly to avoid any penalties related to non-conformance to output formats or runtime errors. It is your responsibility to seek clarification on any aspect of the assignment you find unclear before the deadline.