chaii - Hindi and Tamil Question Answering

(https://www.kaggle.com/c/chaii-hindi-and-tamil-question-answering/overview)

Task

In this assignment you will work on the task of question answering in Indian languages. You will be participating in the ongoing Kaggle competition - https://www.kaggle.com/c/chaii-hindi-and-tamil-question-answering. The last date to register your team (specified in link) for this competition is Nov 8th.

chaii-1 is a question answering dataset in Hindi and Tamil. Given a context and question, the goal of question answering is to predict the answer to the question by selecting a span from the context. Consider the following example from the dataset:

Context:

मानव कंकाल शरीर की आन्तरिक संरचना होती है। यह जन्म के समय 300 हड्डियों से बना होता है और युवावस्था में कुछ हड्डियों के संगलित होने से यह २०६ तक सीमित हो जाती है।[1] तंत्रिका में हड्डियों का द्रव्यमान ३० वर्ष की आयु के लगभग अपने अधिकतम घनत्व पर पहुँचती है। मानव कंकाल को अक्षीय कंकाल और उपांगी कंकाल में विभाजित किया जाता है। अक्षीय कंकाल मेरूदण्ड, पसली पिंजर और खोपड़ी से मिलकर बना होता है। उपांगी कंकाल अक्षीय कंकाल से जुड़ा हुआ होता है तथा अंस मेखला, श्रोणि मेखला और अधः पाद एवं ऊपरी पाद की हड्डियों से मिलकर बना होता है।
मानव कंकाल निम्नलिखित छः कार्य करता है: उपजीवन, गति, रक्षण, रुधिर कणिकाओं का निर्माण, आयनों का भंडारण और अंत: स्रावी विनियमन।
मानव कंकाल अन्य प्रजातियों के समान लैंगिक द्विरूपता नहीं रखता लेकिन मस्तिष्क, दंत विन्यास, लम्बी हड्डियों और श्रोणियों में आकीरिकी के अनुसार अल्प अन्तर होता है। सामान्यतः महिला कंकाल के अवयवों उसी तरह के पुरुषों की की तुलना में कुछ मात्रा में छोटे और कम मजबूत होते हैं। अन्य प्राणियों से भिन्न, मानव पुरुष का लिंग स्तंभास्थि रहित होता है।[2]

सन्दर्भ

श्रेणी:कंकाल तंत्र

Question:

जन्म के समय शिशु के शरीर में कितनी हड्डियाँ होती है?

Answer:

300

Data format

The data provided to you comes in csv format. The dataset will have following columns

- `id` - a unique identifier
- `context` - the text of the Hindi/Tamil sample from which answers should be derived
- `question` - the question, in Hindi/Tamil
- `answer_text` (train only) - the answer to the question (manual annotation) (note: for test, this is what you are attempting to predict)
- `answer_start` (train only) - the starting character in `context` for the answer (determined using substring match during data preparation)
- `language` - whether the text in question is in Tamil or Hindi

Evaluation metric

For each instance in the test set, you are only required to predict the predicted answer_text (and not answer_start). The predictions will be evaluated using Jaccard Index (explained below).

Since all answers are contiguous subsections of the context, your model will internally predict the start and end character (or token) indices that would then be converted to the predicted answer_text by post-processing (which you have to do on your own).

Jaccard index simply calculates the intersection over union of the predicted answer_text tokens (A) and the gold answer_text tokens (B) as follows:

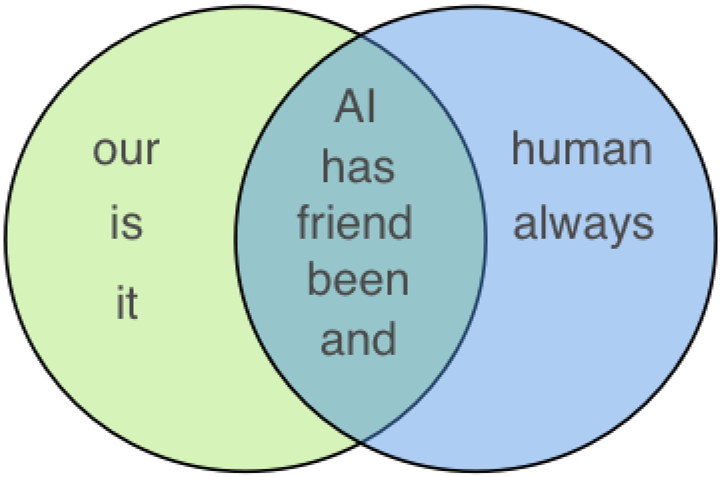
$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

For example, consider the following two texts:

Sentence 1: AI is our friend and it has been friendly

Sentence 2: AI and humans have always been friendly

The Jaccard index will first find the intersection of the tokens present in the two sentences. The number of tokens present in the intersection will then be divided by the no. of tokens present in the union of the two sentences.



we get Jaccard similarity of 5/(5+3+2) = 0.5

For submission format and more details on the evaluation metric, check https://www.kaggle.com/c/chaii-hindi-and-tamil-question-answering/overview/evaluation

Possible Solutions

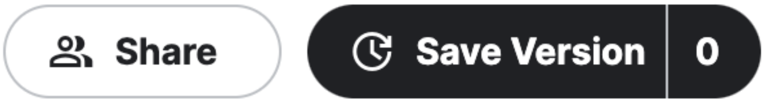
To solve this task, you can apply sequence labeling with multilingual pre-trained models such as mBERT or XLM-Roberta. However, you may want to additionally increase the language-specific data by 1. augmenting the dataset with English question-answer pairs from SQuAD, or 2. translating them from English to Hindi/Tamil by training Machine Translation systems.

You may also take advantage of ensembling methods to boost performance. The SQuAD leaderboard (https://rajpurkar.github.io/SQuAD-explorer/) can provide further hints on training high quality QA systems.

Submission

Chaii is a kaggle code competition. This means that submissions to this competition must be made through Notebooks. Your submission notebook should generate a submission.csv file following the format described above.

The notebooks can be saved by making a commit (save version button on upper right corner)



In order for the "Submit" button to be active after a commit, the following conditions must be met:

- CPU Notebook <= 5 hours run-time
- GPU Notebook <= 5 hours run-time
- Internet access disabled
- Freely & publicly available external data is allowed, including pre-trained models
- Submission file must be named `submission.csv`

Apart from submitting the Kaggle notebook to the competition, we will also ask you to share the Kaggle notebook that replicates your final validation score.

What is allowed? What is not?

General

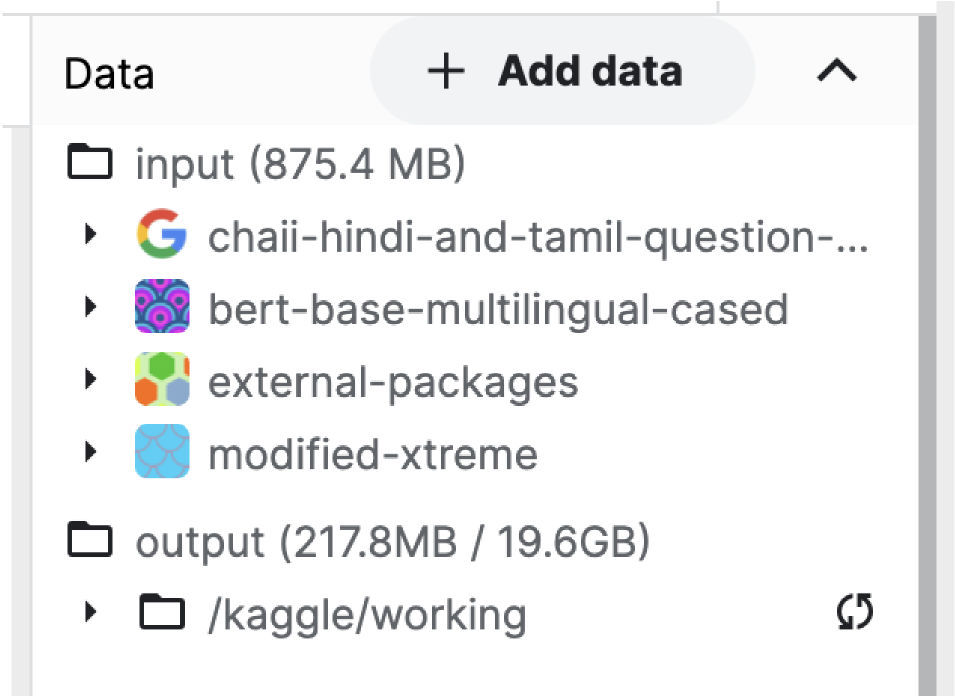
1. The assignment is to be done in teams of two.
2. You must use Python for this assignment.
3. You must not discuss this assignment with anyone outside the class. Make sure you mention the names in your write-up in case you discuss with anyone from within the class outside your team. Please read the academic integrity guidelines on the course home page and follow them carefully.
4. We will run plagiarism detection software. Any team found guilty will be awarded a suitable penalty as per IIT rules.
5. Your code will be automatically evaluated. You get a significant penalty if it does not conform to output guidelines. Make sure it satisfies the format checker before you submit it.

Specific:

This is a no-holds-barred assignment! You are allowed to use any pre-trained model or external data as long as it is allowed by the competition rules (link). You are allowed to use the starter code (link1 and link2) provided by the competition and some publicly available frameworks like Pytorch Lightning, HuggingFace, Fairseq and Allennlp. However, you cannot use any other Kaggle kernels/Github repositories. You have to write all the code on your own (which will be verified using code match softwares).

Kaggle: Using external data and pre-trained models

You can add publicly available external data and pre-trained models to the input folder from the add data button in the folder section. Once committed, the extra data you added will remain in the input folder of your notebook.

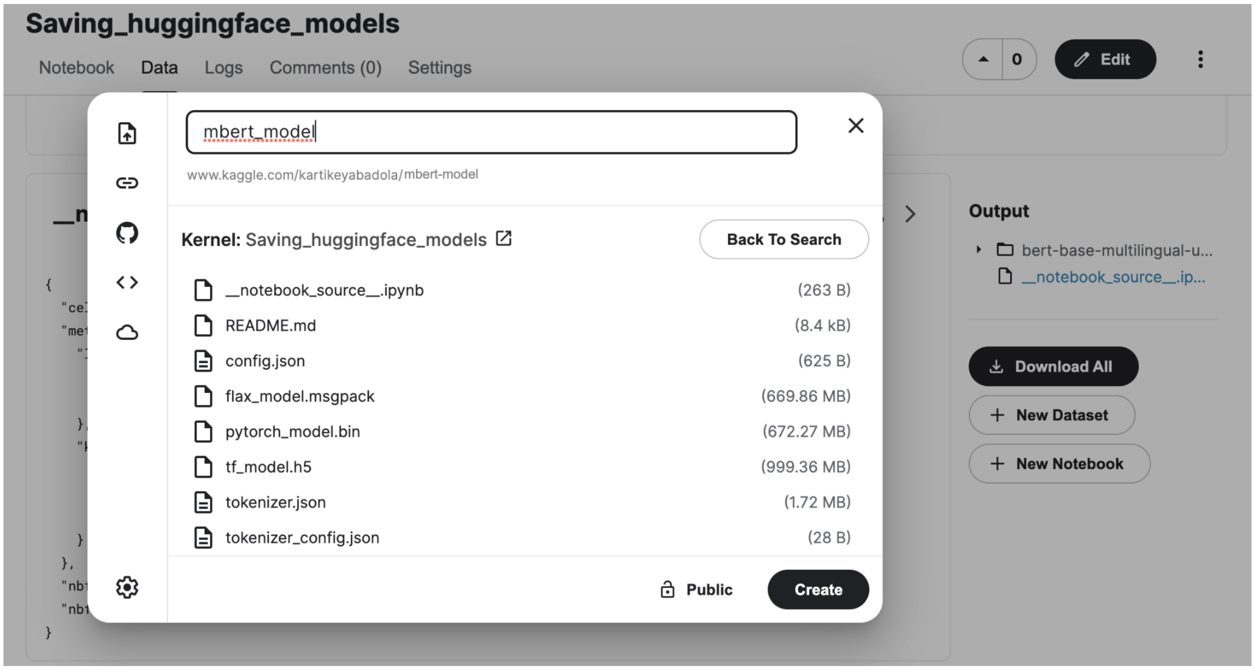


input (875.4 MB)
- chaii-hindi-and-tamil-question-...
- bert-base-multilingual-cased
- external-packages
- modified-xtreme

output (217.8MB / 19.6GB)
- /kaggle/working

For using any Huggingface pre-trained model, you might have to create your own public dataset which you can then add to the input folder. Follow all the steps given in the notebook linked below and keep the notebook running:
https://www.kaggle.com/kartikeyabadola/saving-huggingface-models

View your commit and the saved output would be visible on the right-hand side. Click on new dataset and give your dataset a name. Make sure that the lock below shows public.



Saving_huggingface_models

Notebook | Data | Logs | Comments (0) | Settings

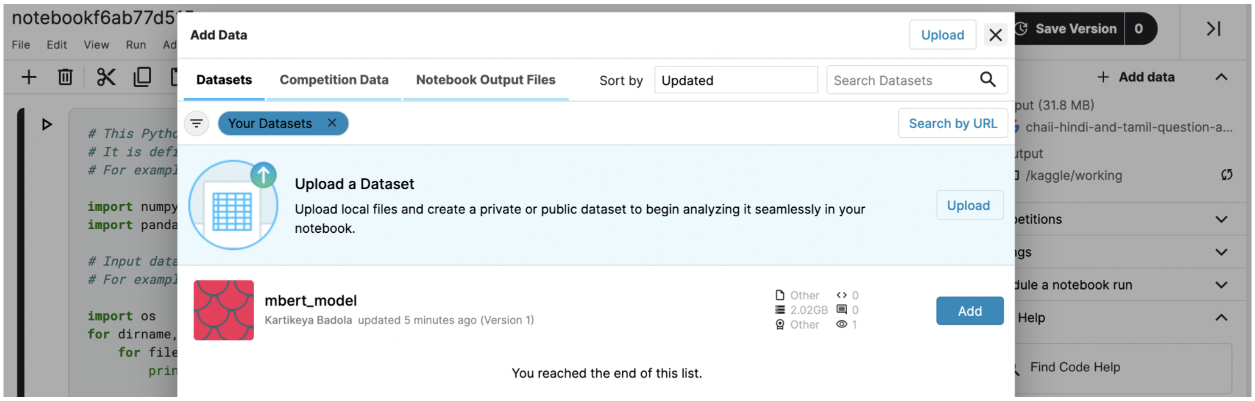
mbert_model

www.kaggle.com/kartikeyabadola/mbert-model

Kernel: Saving_huggingface_models

__notebook_source__.ipynb	(263 B)
README.md	(8.4 kB)
config.json	(625 B)
flax_model.msgpack	(669.86 MB)
pytorch_model.bin	(672.27 MB)
tf_model.h5	(999.36 MB)
tokenizer.json	(1.72 MB)
tokenizer_config.json	(28 B)

Coming back to the notebook you made for the chaii competition, click on add data and add the dataset which you just created.



Add Data

Upload a Dataset

Upload local files and create a private or public dataset to begin analyzing it seamlessly in your notebook.

mbert_model

Kartikeya Badola updated 5 minutes ago (Version 1)

Other · 2.02GB · Other · 0 · 0 · 1

You reached the end of this list.

Kaggle: Installing external libraries

You can find the wheels for many of the external packages from https://www.kaggle.com/deeplearning10/external-packages

Add this as a dataset in your notebook as described in the previous step.

Eg

```
cd /kaggle/working/chaii-packages
cp /kaggle/input/external-packages/* /kaggle/working/chaii-packages
pip install . --no-index --find-links /kaggle/working/chaii-packages/
```