CA6 for NLP Course
School of Electrical and Computer Engineering, University of Tehran

Collaboration You are free to work on this project in teams of two (encouraged) or individually. Individual projects can be less ambitious but should not be less complete: a half-implemented system does not make a good project outcome. All partners should contribute equally to the submission, and all partners will receive the same grade for it. You are also free to discuss your project with others in the course, though only the people on your team should contribute to the actual implementation/experimentation involved. Any external resources used must be clearly cited.
Email: borhanifardz@ut.ac.ir, borhanifard.zeinab@gmail.com

1. Question Answering (QA) Project

Assignment

Your project includes the implementation of a BERT-based model which returns "an answer", given a user question and a passage which includes the answer of the question. For this question answering task, You should use the Persian datasets like SQuAD. You start with the BERT-base pretrained model "bert-base-parsbert-uncased" and fine-tune it to have a question answering task.

What is SQuAD?

Stanford Question Answering Dataset (SQuAD) is a reading comprehension dataset, consisting of questions posed by crowdworkers on a set of Wikipedia articles, where the answer to every question is a segment of text, or span, from the corresponding reading passage, or the question might be unanswerable.

Question Answering on Persian available dataset like SQuAD

The standard final project is to work on modifying and extending a neural question answering system. You should use available Persian **BERT** (**ParsBERT**) and **ALBERT** (ALBERT-Persian). You are given three datasets as part of the Machine Reading for Question Answering (MRQA):
PQuAD, PersianQA, **ParSQuAD**

All three datasets ask questions about Wikipedia articles.

Your goal in this project is to experiment with some improvement to question answering that may allow it to work better.

You should Train **4** Question answering model individually for both **BERT** and **ALBERT** model:

1. PQuAD QA model (use PQuAD dataset)
2. PersianQA QA model (use PersianQA dataset)
3. ParSQuAD QA model (use ParSQuAD dataset)
4. PQuAD and PersianQA QA model (use mix of PQuAD and PersianQA datasets)

Dataset

We have three Reading comprehension datasets like SQuAD

PQuAD, PersianQA, ParSQuAD

The details of dataset is presented in Table1.

Tabe1: details of dataset for ODQA

Dataset	Link	Source
PQuAD	Paper	<<pquad_public.rar>>
PersianQA	Github	<<PersianQA.rar>>
ParSQuAD	Github	<<ParSQuAD.zip>>

Deliverables and Grading

Grading

Code

You should submit any code you wrote on the project due date, this is for documentary purposes only. Please do not include large data files or external resources you used that might be needed to execute it.

Final Report

It should begin with an abstract and introduction, clearly describe the proposed idea or exploration, present technical details, give results, compare to baselines, provide analysis and discussion of the results, and cite any sources you used. This paper should be at least 3 pages excluding references. Your project is not graded solely on the basis of results. You should approach the work in such a way that success isn't all-or-nothing. You should be able to show results, describe some successes, and analyze why things worked or didn't work beyond "my code errored out." Think about structuring your work in a few phases so that even if everything you set out to do isn't successful, you've at least gotten something working, run some experiments, and gotten some kind of results to report.

Grading

We will grade the projects according to the following rubric:

Implementation (35 points)

Is the implementation described reasonable? Is the idea itself technically sound? You might lose points here if we perceive there to be a technical error in your approach. For example, perhaps you added a module to the neural network that leads to no performance change, but it's because it mathematically led to no change in the model due a conceptual error.

Results/Analysis (25 points)

Whether the results are positive or negative, try to motivate them by providing examples and analysis. If things worked, what types of errors are reduced? If things didn't work, why might that be? What aspects of the data/model might not be right? There are a few things you should report here:

- **Key results:** You should report results from a baseline approach as well as your "best" model.
- **Ablations:** If you tried several things, analyze the contribution from each one. These should be minimal changes to the same system; try running things with just one aspect different in order to assess how important that aspect is.

Clarity/Writing (15 points)

Your document should clearly convey a core idea/hypothesis, describe how you tested it/what you built, and situate it with respect to related work as best you can.

- **Abstract and Introduction**: Did you provide a clear summary of the motivation, methodology, and results?

- **Method:** Is the presentation of what was done clear?
- **Results:** Is the work experimentally evaluated? Are there clear graphs and tables to communicate the results? make your analysis more detailed than that.

2. **Natural Language Understanding Projects (45 points)**

The task-oriented dialogue system is the basis of virtual assistants like Alexa, Siri, Cortana, and Portal has been increasingly used in modern society; users interact with them across different domains to complete diverse tasks and achieve their specific goals.

A key component of these task-oriented dialogue systems is Natural Language Understanding (NLU) which aims to derive the intent of users and fill the value for the slots of the utterance. For example, in the utterance "Play a chant by Mj Cole", a dialogue system should correctly identify that the user's intention is to give a command to play a song, and that Mj Cole is the artist name that the user would like to listen.

MASSIVE is a parallel dataset of > 1M utterances across 51 languages with annotations for the Natural Language Understanding tasks of intent prediction and slot annotation. Utterances span 60 intents and include 55 slot types.

In As part of this project, you will develop a model for intent detection and slot filling on a MASSIVE dataset in Farsi. With the label partition in the dataset, you should partition data into training, testing, and development.

It is important to report statistics of the persian dataset that include counts of intents and slot types in MASSIVE. To model, any transformer and network can be used. The slot filing performance should be evaluated by reporting the F1 score and intent prediction with accuracy.

The rules for the final report, code, and grading are the same as those for the above project.