CSE 441 Software Engineering

Spring 2020

Project Abstract

Saumitra Yadav (201507647)
Sunil Gundapu (2018701022)
Prashant Kodali (2018801011)
Neha Motlani (20161004)
Ananya Mukherjee (2018801009)
Hiranmai Sri Adibhatla (2018900044)

Project Name: NLP toolkit as a service

Problem Statement:

In the course of any NLP application development, there are many basic steps which are common across all NLP tools. These may be as basic as tokenization, Named Entity Recognition (NER), and extend to as complex tools as Sentiment Analysis, Machine Translation etc.

As different teams within an enterprise look at developing their own NLP applications, the initial parts of all their pipelines have same building blocks: sentence tokenizers, word tokenizers, NERs, parsers etc.

Further, as we get deeper into these pipelines, we see complex tools: like sentiment analysis, Language Models or Machine Translation. In the context of resource poor languages, these complex tools are all the more important because of:

1. Pivoting: translate a low resource language to resource rich language (like English) and use the rich toolkit of resource rich language.
2. As you look at complex applications at discourse or pragmatics level, you need reliable outputs from these tools like sentiment analysis.
3. Provide baselines.
4. A shared knowledge repo for all implementations, allowing various teams to have look at the existing work, tools available to them, evaluated by other teams. So that multiple teams do not spend time in going through similar exercise.

Now, it is counter-productive for all these teams which are working on their individual applications to write these tools on their own, and here lies the scope to cut down redundant work across teams. This maybe as basic as a tokenizer or as complex as MT.

Proposed Solution:

A web portal that makes use of the underlying APIs and

a. allows users to look at the sample outputs by keying in their inputs;
b. also be capable of allowing users to upload documents and get the processed output of the whole document;
c. APIs callable from any script. User needn't just rely on the portal and its UI

The scope of tools that will integrated in the portal would be:

1. Sentence and word Tokenization
    a. Normal text
    b. Social media text
2. Transliteration
3. Named Entity Recognition
4. Summarization: a naïve k-means based summarization
5. Transformer based LM: output and representations
6. Sentiment analysis
    a. English
    b. Code mixed: English and Hindi. Hinglish
7. Machine Translation
    a. For a specific language pair
    b. Also give performance measures.

The solution should be modular enough so that teams can integrate their own tools to this portal. Allowing this portal to grow in a wide and decentralized manner.