Project Guidelines - INLP

March 2024

General Instructions

1. This is a group project. Each group (team) will consist of 2 to 3 members only.

2. A mentor will be announced for each project. Please make sure to stay in touch with the mentor for any clarifications needed for the project.

3. Feel free to approach TAs if anything seems difficult to grasp. Make use of the TA Office Hours. Ask for tutorials on any common topics of concern if necessary.

4. The deadlines have been planned to allow adequate duration for each submission. Hence, divide the project into deliverables and plan their completion accordingly.

5. When including code in the submission (whenever applicable), it is necessary to include a README file explaining how the code has been divided into files and instructions for their execution, along with a requirements.txt file containing the list of necessary dependencies.

6. Please ensure there's no plagiarism for the parts that you have to implement. It's okay to use existing code for baselines and supplementary code that you'll be needing. However, your own implementation must be based on your understanding.

7. Any detected case of plagiarism would result in strict disciplinary actions, which could also include an F grade for the course, apart from 0 being awarded for the project component.

On the Implementation

1. This document has tried to scope each project and your expectations from it to establish a concrete ground. You are free to make any reasonable changes to the scope after discussion with the mentor & TAs.

2. Please start the implementation part of the project early after getting a grasp of the literature and the ideas involved. One typically keeps running into bugs, runtime errors which take days to solve.

Project Submissions

There will be three submissions for the project, each bearing a considerable weight for the overall project marks.

1. Project Outline

Deadline: March 11, 2024, 23:59

Weight: 15% of Project

Here, you will be explaining the problem that you will solve through the project. Describe the scope of the project properly. Explore the datasets available and their feasibility and metrics for evaluation, and write about your findings. Go through a few relevant papers and include a short literature review for the problem. This will help you identify the baselines you can use and grasp the research area. Plan the implementation of the project over the period till final submission and include a tentative timeline for the same. The overall length of the report is expected to be around 3-4 pages. Submit the report PDF, which should be named in the format <TeamNo>-Outline.pdf.

Apart from this, it will be helpful if you go through an implementation of the approach available on GitHub and try running it to see its reproducibility. Not for the report, but this can help you with the later submissions.

2. Interim Submission

Deadline: April 4, 2024, 23:59

Weight: 25% of Project

This checkpoint is to see if you're on track with the project timeline. By this checkpoint, it is expected that you have done some exploratory analysis using your dataset and checked the performance against the baselines that you'll be using. Remember that you're not expected to implement the baselines from scratch, and it only helps if you add more baselines for evaluating your final approach. You will thus also have an evaluation pipeline ready to check your model's performance.

Simultaneously, start implementing the approach that you're supposed to. Include all your progress in the report. Submit the code containing your implementation so far along with the report in a .zip file, which should be named in the format <TeamNo>-Interim.zip.

3. Final Submission

Deadline: May 6, 2024, 23:59

Weight: 60% of Project

For the final submission, you should have completed the implementation along with sufficient analysis of results. This includes quantitative analysis showing the performance over the metrics and qualitative analysis demonstrating the expected output for a few test cases. As part of the final report, start with the introduction of the problem, explain the selection of appropriate baselines from the literature study, dataset characteristics, your understanding & interpretation of the approach being implemented and a summary of your implementation. Importantly, along with the results, also include an analysis section at the end describing your findings - what works better and why, and what does not work.

Along with the report, also include a well-designed, well-illustrated presentation which would include all the points covered above. Note that the presentation content should not be directly copy-pasted from the report and should be presented properly. This will be used for the project walkthrough during evaluation.

The final submission will consist of these three components:

a. Code folder

Since you will be implementing a SoTA approach on a problem, it will be good to also showcase the same on GitHub.

b. Report

c. Presentation

Include all these deliverables in a .zip file named in the format <TeamNo>-Final.zip.

We have provided a list of 14 project topics along with their description & expected scope below.

1. **Semantic Textual Similarity**

Semantic Textual Similarity (STS) measures the degree of equivalence in the underlying semantics of paired snippets of text. Given two sentences, the model should return a continuous-valued similarity score on a scale from 0 to 5, with 0 indicating that the semantics of the sentences are completely independent and 5 signifying semantic equivalence. Performance is assessed by computing the Pearson

correlation between machine-assigned semantic similarity scores and human judgments.

Work on Cross Lingual Model (Spanish - English) or only English – English?

Dataset:

STS 2017 Cross-lingual English-Spanish

Data STS 2017 Trial Data

STS 2017 Evaluation Sets v1.1

More Information and related Datasets can be found in at Wiki: STS Wiki

2. **Extracting Keyphrases and Relations from Scientific Publications**

The task deals with the extraction of key phrases automatically given a scientific publication. Moreover, the key phrase needs to be labeled and should be related to other key phrases. PROCESS, TASK, and MATERIAL form the fundamental objects in scientific works. Scientific research and practice are founded upon gaining, maintaining, and understanding the body of existing scientific work in specific areas related to such fundamental objects. This task aims to address the related fundamental problems in the field.

Corpus Description: https://scienceie.github.io/resources.html

3. **Measure Text Fluency**

Fluency is commonly considered as one of the dimensions of text quality of MT. Fluency measures the quality of the generated text (e.g., the target translated sentence) without taking the source into account. It accounts for criteria such as grammar, spelling, choice of words, and style. A typical scale used to measure fluency is based on the question, “Is the language in the output fluent?”.

Reference: **https://ieeexplore.ieee.org/document/1244655?arnumber=1244655**
https://aclanthology.org/K18-1031.pdf

4. **Words Sense Disambiguation**

The automatic understanding of the meaning of text has been a major goal of research in computational linguistics and related areas for several decades. The task of Word Sense Disambiguation (WSD) consists of associating words in context with their most suitable entry in a pre-defined sense inventory. The de-facto sense

inventory for English in WSD is WordNet. For example, given the word “mouse” and the following sentence:

“A mouse consists of an object held in one's hand, with one or more buttons.”
we would assign “mouse” with its electronic device sense (the 4th sense in the

WordNet sense inventory).

Typically, there are two kinds of approaches for WSD: supervised (which make use of sense-annotated training data) and knowledge-based (which make use of the properties of lexical resources).

Supervised: The most widely used training corpus is SemCor, with 226,036 sense annotations from 352 documents manually annotated. All supervised systems in the evaluation table are trained on SemCor. Some supervised methods, particularly neural architectures, usually employ the SemEval 2007 dataset as a development set (marked by *). The most usual baseline is the Most Frequent Sense (MFS) heuristic, which selects for each target word the most frequent sense in the training data.

Knowledge-based: Knowledge-based systems usually exploit WordNet or BabelNet as a semantic network. The first sense given by the underlying sense inventory (i.e., WordNet 3.0) is included as a baseline.

5. **Hypernym Discovery**

Hypernymy, i.e., the capability for generalization, lies at the core of human cognition. Unsurprisingly, identifying hypernymy relations has been pursued in NLP for approximately the last two decades, as successfully identifying this lexical relation contributes to improvements in Question Answering applications (Prager et al. 2008; Yahya et al. 2013) and Textual Entailment or Semantic Search systems (Hoffart et al. 2014; Roller and Erk 2016). In addition, hypernymic *(is- a)* relations are the backbone of almost any **ontology**, **semantic network,** and **taxonomy** (Yu et al. 2015; Wang et al. 2017), the latter being a useful resource for downstream tasks such as web retrieval, website navigation or records management (Bordea et al. 2015).

For each subtask and setting, we provide a list of input terms (hyponyms) as well as a large *vocabulary* extracted from each corpus. The team is expected to deliver, for each input term, a *ranked list of candidate hypernyms* (up to **15**) from the provided vocabulary.

Corpus:
https://drive.google.com/file/d/14_RgB3_it7a_1mLXeRCyzwY5BHdWgnlP/view

General-purpose corpora. For the first subtask we use the 3-billion-word **UMBC corpus** (Han et al. 2013), which is a corpus composed of paragraphs extracted from the web as part of the Stanford WebBase Project. This is a very large corpus containing information from different domains.

6. **Neural Unsupervised Paraphrasing**

Paraphrasing is expressing a sentence using different words while maintaining the meaning. In this project teams will be implementing unsupervised approaches to generate paraphrases for Indian Languages.

7. **Anaphora & Coreference Resolution**

In discourse, anaphora may be defined as a reference back to a word used earlier in a text or conversation, to avoid repetition. In this task, teams are expected to create an algorithm that resolves anaphora based on the dataset chosen.

In discourse, coreference may be defined as the phenomenon when two or more expressions in a text refer to the same person or thing; they have the same referent. In this task, teams are expected to create an algorithm that resolves coreference based on the dataset chosen.

8. **Natural Language Inference**

A fundamental phenomenon of natural language is the variability of semantic expression, where the same meaning can be expressed by or inferred from different texts. Textual entailment recognition is the task of deciding, given two text fragments, whether the meaning of one text is entailed (can be inferred) from another text. Given two text fragments, one named text (t) and the other named hypothesis (h), respectively. The task consists in recognizing whether the hypothesis can be inferred from the text. TE has a three-class balanced classification problem over sentence pairs:

1. Contradiction
2. Entailment
3. Neutral

Dataset: multinli , SICK , SNLI

9. **Neural Dependency Parser**

Dependency parsing is a popular grammar formalism used for better understanding a sentence structure. A dependency parsing mechanism can give a parsed dependency tree for a given sentence, which involves a set of relations over its words such that one is a 'dependent' of the other. Training such dependency parsers involves making use of annotated data to learn the way these trees are constructed, which can be modeled to give good performance by effectively using neural networks. Based on the approach taken to model the problem, the parsing can be either **transition-based or graph-based**. In this project, the students will implement a neural model which will be trained to perform dependency parsing on a sentence, using a method of their choice.

Resources:

Universal Dependencies Project
Non-Projective Dependency Parsing in Expected Linear Time
Stanford's Graph-based Neural Dependency Parser at the CoNLL 2017 Shared Task

[OPTIONAL]

Additionally, while a lot of literature exists in the dependency parser community in NLP, there are few well developed tools for Indian languages. Some basic reading material for the same:

1. https://www.researchgate.net/publication/242103292_Bidirectional_Dependency_Parser_ for_Hindi_Telugu_and_Bangla
2. https://www.aclweb.org/anthology/W12-5617.pdf

Languages: Bengali, Telugu, Marathi, Tamil, Hindi

10. **Textual Coherence**

An important aspect for a textual discourse, coherence measures the readability, clarity, and consistency of ideas expressed in a passage. Within a discourse, the coherency is exhibited at both - local and global levels. Whether to model the problem on a global level or to decompose it into a series of local decisions remains a modeling choice. In this project, students will experiment with neural models in measuring textual coherence.

Resources:

A Cross-Domain Transferable Neural Coherence

Model Neural Net Models of Open-domain

Discourse Coherence

11. **Semantic Role Labeling**

Semantic Role Labeling is a task based on identifying the role of a word or a group of words in a sentence. Semantic role labeling is a very useful idea and is used intrinsically in a number of downstream tasks such as question-answering, inference, knowledge graph creation and participant detection.

The aim of this project is to develop a linguistically grounded semantic role labeler based on the Hindi Dependency Treebank and Hindi PropBank. Two previous attempts at this task were:

https://www.aclweb.org/anthology/L16-1727.pdf

http://lrec-conf.org/workshops/lrec2018/W29/pdf/28_W29.pdf

These can be used as references for the theory, but the idea is to be able to implement a model (statistical or neural) that analyzes a sentence based on the role of the individual participants, from a combination of the part of speech, the dependency label and other linguistic information useful in determining theta role and associated information.

Languages: Bengali, Tamil, Telugu

12. **Contextual Embedding (ElMO) for Indian Languages**

In this project, teams are expected to implement the Contextual embeddings ElMO on cleaned data in any Indian language. While the algorithms for these are available, evaluation metrics for word representations in Indian languages are far and few between. The teams are expected to implement similarity and analogy as well and provide how well the embeddings fare.

Languages: Hindi, Telugu, Bengali, Marathi, Tamil

13. **Code Mix Generation**

Code Mixing is a phenomenon where a speaker mixes two or more languages in a single sentence, typically occurring in a bilingual/multilingual society. Hinglish is an

example of code mixing. This is more frequently seen on user generated text on social media, comments on websites etc. The main objective of this project is to generate code mixed text using statistical and simple neural algorithms.

Languages: Hindi

14. **Question Answering**

Question Answering (QA) is a subfield of Natural Language Processing (NLP) that focuses on automatically answering questions posed in natural language. The goal of a QA system is to understand the questions asked and provide a concise and relevant answer.

The process of building a QA system involves several steps, including understanding the question, retrieving relevant information, and ranking the answers based on their relevance. The system must also consider the context of the question and use background knowledge to generate the answer.

QA systems have many practical applications, including knowledge management, customer service, and education. They can help reduce the time and effort required to find answers to questions and provide more accurate and relevant answers compared to traditional search engines.

Daasets: SQuAD, MS MARCO, BioASQ, TREC QA, Natural Questions (Google AI), NarrativeQA, HotpotQA

15. **Your own Idea**: You are free to suggest project ideas of your own. Make sure to properly define your project idea for it to be approved.