**1) Algorithm-Focused Tasks**

You will be an **algorithm engineer** who designs, analyzes, and implements algorithms to solve specific problems. Our products are built upon algorithms that support and automate knowledge acquisition and decision making-- i.e., what humans in their productivity every day. Such algorithms are often natural language processing (NLP) or data processing in nature, to embed artificial intelligence into our products.

As an algorithm engineer, you will be familiar with or can quickly learn the state of the art of computer science in general and artificial intelligence in particular. You will be comfortable in finding and reading research papers, identifying cutting-edge techniques, and adopt or adapt them to our products.

For this task, you can choose any of the following NLP problem:

* **Problem 1: Named entity recognition (NER)** -- to identify entities mentioned in text. Can we recognize and extract entities in a text document-- what are the person, location, organization and other entities mentioned?

**Task A. Survey, Implement, Apply Algorithms for a Task**

* **Task:** You can choose any of the above problems. Let's try to survey the problem, implement a solution, and apply it to an application setting.
* **Part 1: Survey.** For the problem that you choose, survey about the problem and learn the research that has been done for it. Go to Google Scholar and find a recent/good survey paper about the problem from which you can learn the related research.

**Q1:** *Can you summarize the overall research and state of the art you learned in no more than one page?*

* **Part 2: Implementation.** Can you try to **implement (write your own code) one technique** from a paper you have read on the problem? Find a recent (within 5 years) **original research paper** and implement the specific techniques that it proposes. Please note: Our purpose here is to practice implementation, so please choose a paper that is interesting and non-trivial and try to build/reproduce its techniques. You can use any existing libraries/tools, but you should implement the main logic/algorithms and train the models as the paper proposes.

**Q2:** *Why do you choose this paper? Can you present your implementation and how it works? Are you able to reproduce the results of the paper? Observe, analyze and interpret the results.*

* **Part 3: Application.** Let's apply the problem to help users read webpages. You will create a [Chrome extension](https://developer.chrome.com/docs/extensions/mv3/getstarted/):
  + When a user visits a webpage in Chrome, your extension will read the content of the page, send it to a server (which should be your implementation in Part 2).
  + Your server will apply the problem on the content (e.g., extract entities if Problem 1) and send the results (e.g., extracted entities) back to the extension.
  + The extension will then display the results in the webpage. How? You can decide-- just make your extension useful.
  + Reference: [Connecting Chrome Extension with Python Backend](https://medium.com/@oaishi.faria/connecting-chrome-extension-with-python-backend-912d1d0db26)

**Q3:** *Can you present your extension and how it works? Why is it useful? Any ideas to make it even more useful?*

* **Part 4:**
  + **Q4.1:** *What have you learned from the exercise above?*
  + **Q4.2:** *Do you think the results were satisfactory for practical usage in the application of browser extension? How can we adapt / improve the techniques to improve?*
  + **Q4.3:** *What new applications can be built using this problem? Propose a cool application you can build that does not exist today.*

# Problem

**Named entity recognition (NER)** -- to identify entities mentioned in text. Can we recognize and extract entities in a text document-- what are the person, location, organization and other entities mentioned?

# Survey

Papers for Reference:

* 1. Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li

A Survey on Deep learning for Named Entity Recognition

2018

* 1. Vikas Yadav, Steven Bethard

A Survey on Recent Advances in Named Entity Recognition

2019

* 1. Jason P.C Chiu, Eric Nichols

Named Entity Recognition with Bidirectional LSTM-CNNs

2015

NER Dataset:

* CoNLL 2003
* OntoNotes

System:

* **Knowledge-based/Rule-based systems:**

Design based on lexicon resources, domain specific knowledge, dictionary, syntactic lexical.

* **Unsupervised Learning:**

A typical approach of unsupervised learning is clustering. Clustering-based NER systems extract named entities from the clustered groups based on context similarity.

* **Feature-engineered supervised systems:**

Applying supervised learning, NER is cast to a multi-class classification or sequence labeling task.

* **Feature-inferring neural network systems/ Deep learning:**

Modern neural architectures for NER can be broadly classified into categories depending upon their representation of the words in a sentence. For example, representations may be based on words, characters, other sub-word units or any combination of these.

Evaluation Metrics:

Exact match / Relax match:

* Precision/Recall
* F-Score

Relaxed F1 considers a prediction to be correct as long as part of the named entity is identified correctly. Strict F1 requires the character offsets of a prediction and the human annotation to match exactly.

Table show recent research systems and its result.

A table of information

Description automatically generated

# Implementation

The implementation follows research Jason P.C Chiu, Eric Nichols, *Named Entity Recognition with Bidirectional LSTM-CNNs.*

Why is this paper?

Get the information from the survey, this paper is chosen due to:

* This paper has the highest F1 score for English dataset.
* This paper has extracted feature combine from word, character and casing.
* Applying neural network for NER task is a state-of-the-art technique, this paper is the good for practice implementation.

The implementation is put on github:

* [Named\_Entity\_Recognition/Named\_Entity\_Recognition](https://github.com/truongthuanr/self-project/blob/main/11_Named_Entity_Recognition/Named_Entity_Recognition.ipynb)

# Application

Q1

Q2

Q3

# Sumarization

Q1

Q2

Q3

# References

A Survey on Deep Learning for Named Entity Recognition