Bartłomiej Kowalczuk - Nanodegree Capstone Project Proposal

Named entity recognition

1. Domain Background

Named entity recognition (NER) is a task of tagging entities in text data, that focuses on classifying real-world entities into predefined categories such as, among others, the names of persons, organization or locations. Approaches for tagging, typically use BIO notation, which differentiates the beginning (B) and the inside (I) of entities. O is used for non-entity tokens - this notation will be used in this project.

In Natural Language Processing NER can highly enhance systems that perform text search and text inference. Knowing the named entities, we can simply extract part of texts that contain them, when initially we didn't know the exact entities. We can improve our and algorithm understanding of everything we know about a person, an organization or a place mentioned in a document.

A great example of usage of an algorithm that can perform NER with high precision would be resume scoring model. Such a model would obtain a resume in unstructured format, process and tag entities which could be followed by some scoring model that would rate it based on features (entities). Another way it could be useful is simply for analytical purposes during recruitment processes, extracted and highlighted entities would speed up the process of resume analysis.

Currently the best algorithms used for named entity recognition can be found here: Named Entity Recognition - paperswithcode. These are mostly transformer based, however for some specific tasks BiLSTM-CRF are still performing better.

2. Problem Statement

Problem can be defined as: classification of tokens (words) in a sentence into multiple classes (using BIO notation) - named entity recognition. For classification Bidirectional LSTM with CRF and DistilBERT will be utilized and compared.

3. Datasets and Inputs

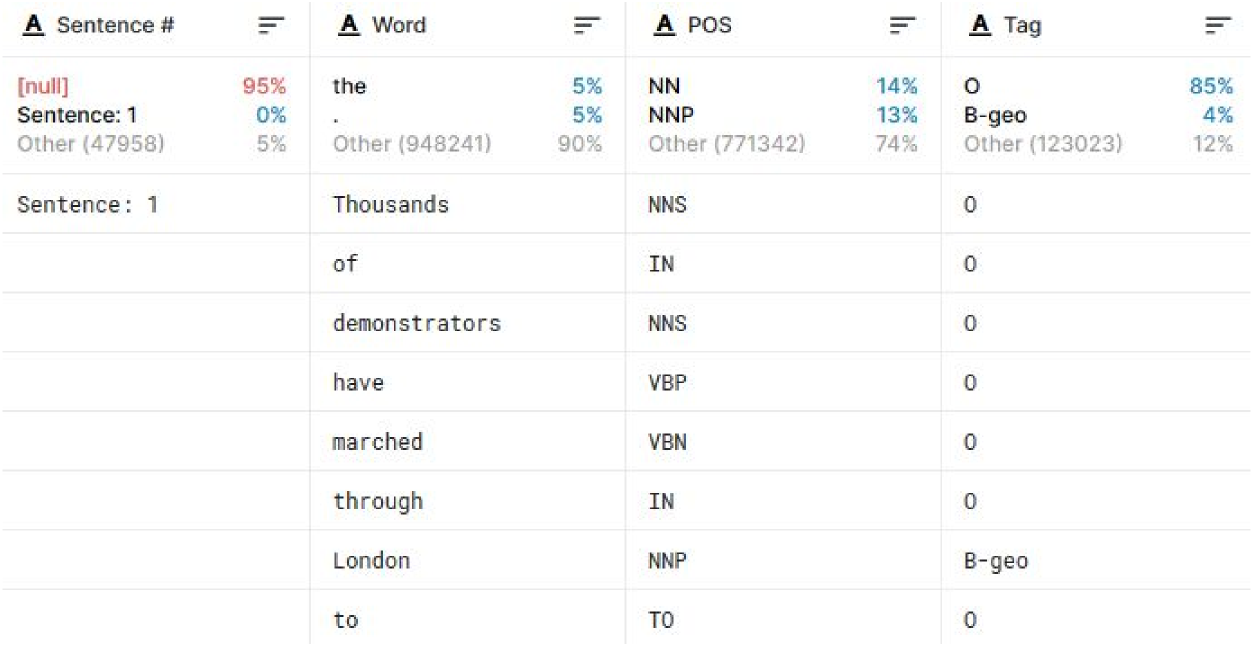
Data source: Annotated Corpus for Named Entity Recognition

This is the extract from Groningen Meaning Bank (GMB) corpus which is tagged, annotated and built specifically to train the classifier to predict named entities such as name, location, etc.

In the dataset there are:

- 47959 sentences
- 1354149 words
- 17 distinct entity tags (names)

Sample of the dataset structure is presented below:



Sentence #	Word	POS	Tag
[null] 95% Sentence: 1 0% Other (47958) 5%	the 5% . 5% Other (948241) 90%	NN 14% NNP 13% Other (771342) 74%	O 85% B-geo 4% Other (123023) 12%
Sentence: 1	Thousands	NNS	O
	of	IN	O
	demonstrators	NNS	O
	have	VBP	O
	marched	VBN	O
	through	IN	O
	London	NNP	B-geo
	to	TO	O

Each token has exactly one label (tag) and sentence number. Column *Word* (aggregated by sentence number) will be used as an input and column *Tag* as an output.

4. Solution Statement

Data will be explored and then transformed for following models:

- Bidirectional LSTM with Conditional Random Field (CRF) layer - Keras
- DistilBERT (fine-tuning) - Hugging Face transformers + Keras

Both models will be tuned and compared on the same train and test sets.

5. Benchmark Model

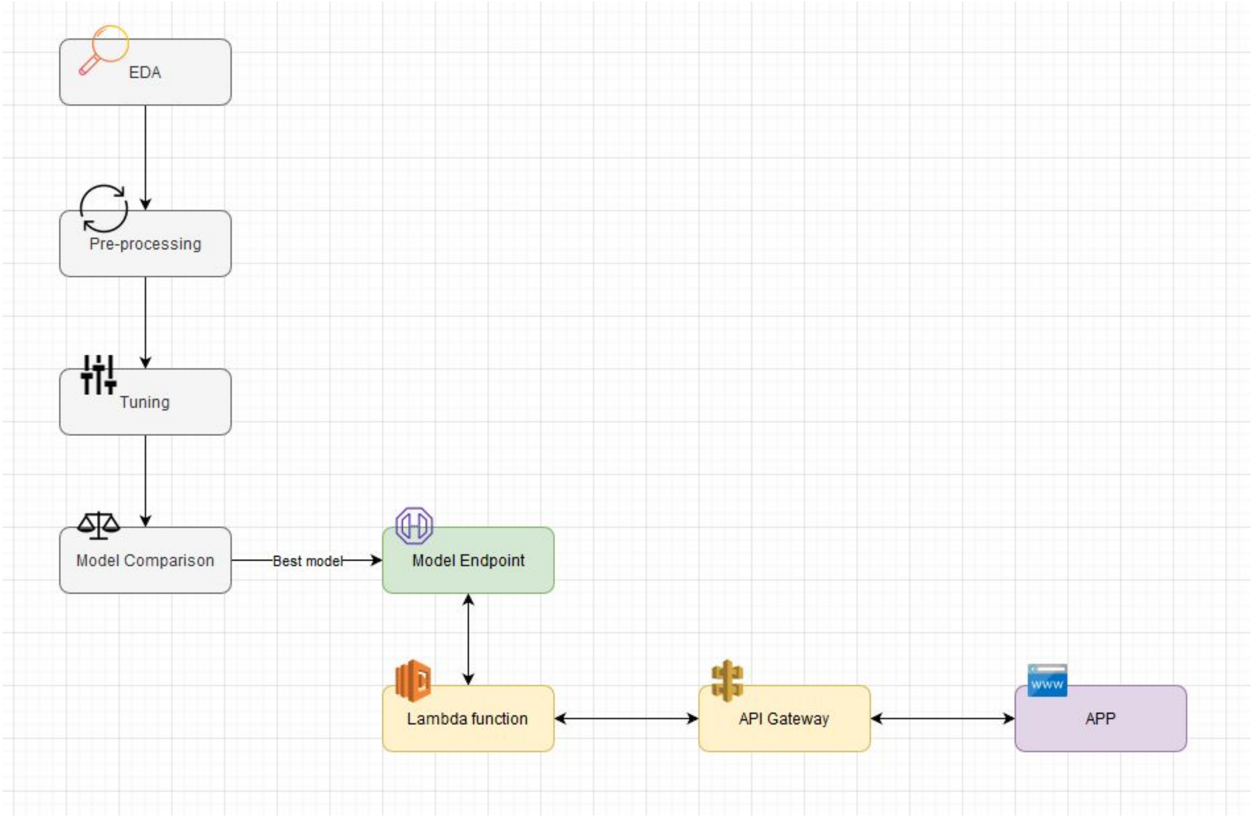
Solutions published on kaggle kernels score around 80-85% F1-Score on their validation set and I would aim at achieving a higher score. I assume that my first proposed model, BiLSTM, would play a role of challenger for DistilBERT.

6. Evaluation Metrics

For evaluation purposes and comparison of models F1-Score would be used as it is a widely used metric in NER solutions.

7. Project Design

Project structure is presented on a diagram below:



First step would be to perform an EDA, then preprocess data for following models. Tune and train models, compare them against each other using F1-Score and deploy the best one. Finally, create a Lambda function that will prepare received input for the model endpoint and will send back the predictions to API Gateway that would send it directly to an app that user operates.