Entity Resolution Homework 3

Francesco Peracchia
1906895
peracchia@studenti.uniroma1.it

Abstract

A task composition made by ambiguous pronoun detection, entity recognition, and finally pronoun-entity aggregation rely on the name Coreference Resolution, this result is a complex and unique task composed by the previous three steps. Is proposed a possible solution for Coreference resolution inspired Gendered Ambiguous Pronouns Shared Task (Attree, 2019)

encoded	label
0	Nothing
1	Entity A
2	Entity B

Table 1: Our classifier is trained to receive in input text, an ambiguous pronoun and a couple of possible referred entities, this is translated in three different inputs for the tokenizer. Referring to these three possible inputs the model assigns scores to them in training and return the most probable labels in inference mode, above how the labels are encoded for both training and successively translating the inferred output to entities into nothing choice.

1 Data processing

In order to solve Coreference resolution between a pronoun and a couple of possible entities each text input in processed following the below proposed sequence of operations.

- Truncate text input up to the last element involved, pronoun ,first or second entity.
- Add pronoun and entities indicators.
- Add the second text query.
- Create three different contextualized queries and tokenization.

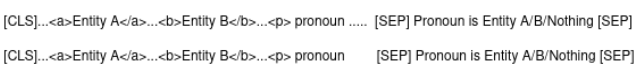

Figure 1: Text is first truncated up to the last element present, then separated by [SEP] token the query is added, this is the input that will be provided to BERT.

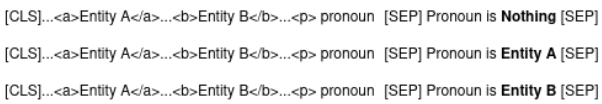

Figure 2: In order to extract a single contextualized representation, for the combined context and query, the pooled output or embedding correspondent to [CLS] token, is extracted for each of the three options.

Text input is used to extract the context, in order to avoid that unuseful information is integrated part of the context, text is cropped exactly after the last the elements is founded into the text. As proposed (Attree, 2019) pronoun and entities indicators are successively used as special tokens to incorporate entities structures information into the contextualized embedded 1, finally three different copies with different queries are added, again as before is re proposed the same strategy indicated by (Attree, 2019), the final result 2 generated for each input is a sequence of three sets of tokens with the same length, and completely equivalent if not for the last option choice.

2 Model

A sentence composed of L words is hence transformed into a vector of length 768, this phase makes use of BERT (Devlin et al., 2018) and different models have been tested, note that since each instance will generate three different inputs; with a batch size equal to N, the final batch size for BERT will be N dot 3, since for each ambiguous text

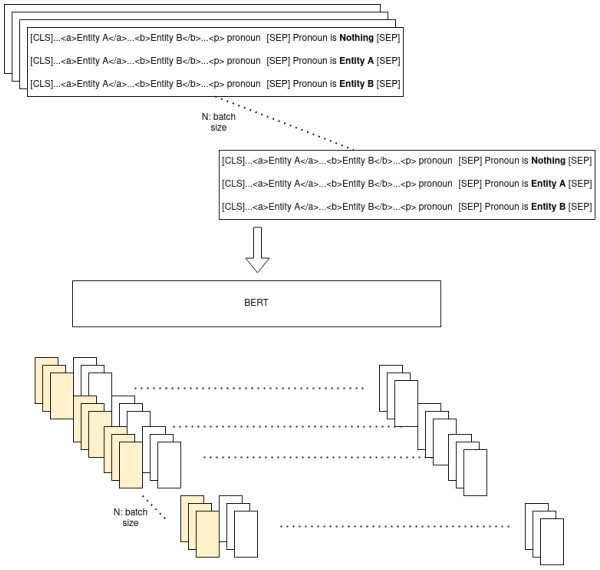

Figure 3: I.e, for batch size equal to 32 are generated instances of length L_{pad} for a total dimension of $32 * 3$, while for ground truth output of 32 elements, long tensor draws elements from 1, note that since BERT input could have different length the [PAD] token is used to pad the input sequence and generate with the tokenizer the desired tokens and attention mask.

are then generated 3 different embeddings, each as pooled contextualized representation of the query proposed 2.

BERT pooled output generates for each text input three different tensors, then is required a strategy that indicates which of these is the final predicted class between "Nothing", "A", "B". A simple MLP is used as a regressor to predict a score, the label with the highest score is the final predicted class. In order to handle both generalization capability, underfitting and overfitting, a tradeoff between MLP's parameters number and regularization techniques should be conveniently balanced, especially are performed some experiments by changing dropout probability inside the MLP.

3 Training

In order to train the model Adam optimizer with $lr = 0.000008$, a scheduler with exponential learning rate decaying has been used, gamma was set to 0.96, this setting results in a fast training since only after 4 epochs we can achieve sufficient accuracy, then in order to avoid overfitting early stopping was implemented and different patience values were used. Another setting for a smoothed training was using dropout to better generalize during the training procedure, the most performative setting was

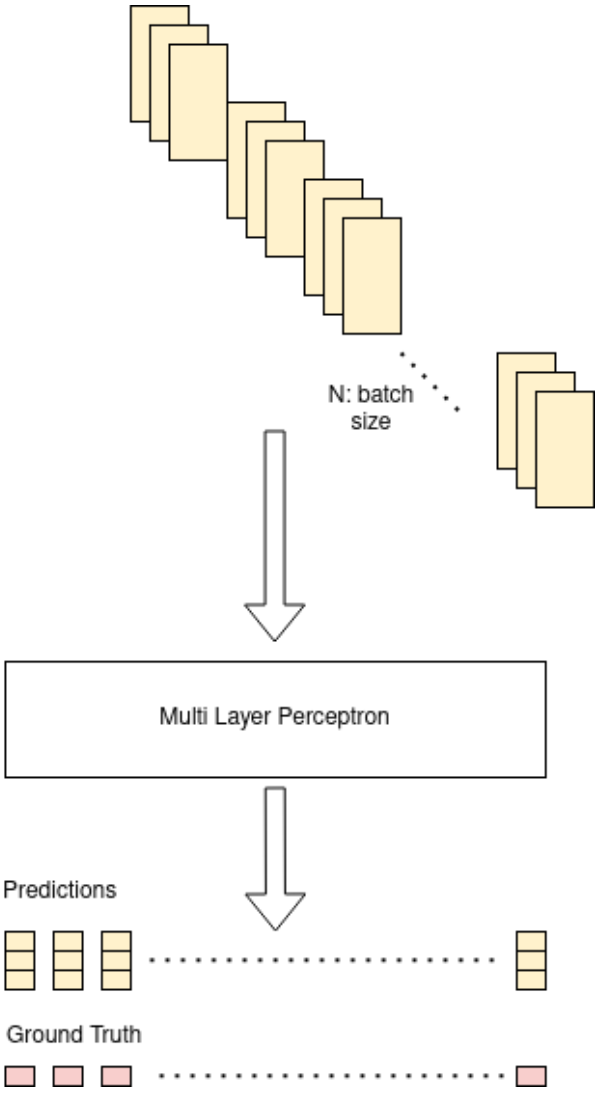

Figure 4: [CLS] embeddings, also called pooled embedding, from all three elements are then used in input to a MLP that assign a score to each input, finally the option between those with the highest score result in the predicted label in inference only the predicted labels is 6+returned while during training all the scores are returned to compute the loss and its successive backpropagation.

with $p = 0.5$ and one only hidden layers, preceded by a batch normalization layer. Finally Cross Entropy is the chosen loss, and in order to prevent exploding gradient gradient clipping was used with hyperparameters $c = 0.6$, this when the gradient computation gets too large rescale it to keep it small enough to proceed thought gradient descent without undesired behaviours. A second model with another hidden layer and RELU as activation function was proposed but it results to be less accurate.

4 Results

Following (Attree, 2019) was possible to generate with BERT, contextualized representation from a context and a query applied over it; from that was extracted a sequence [CLS] pooled embeddings and then using a MLP perceptron were computed scores between all three different cases, "Nothing"," Entity A", "Entity B". Discriminate between both the entities achieve results with F1 score above 0.8 while our model remains fragile in detecting when both the proposed entities are not valuable candidates for being the reference entity, hence f1 score of "Nothing" class pass 0.5 in f1 score. However, it should be remarked fig.5 where only 62 instances are available for this class in the evaluation and in a similar proportion in training. Then in order to extend our results and to understand how to improve the accuracy of these labels it could be interesting to combine this task with entity identification, this could be used to feed the MLP only when we are sure that it is referring to at least one entity.

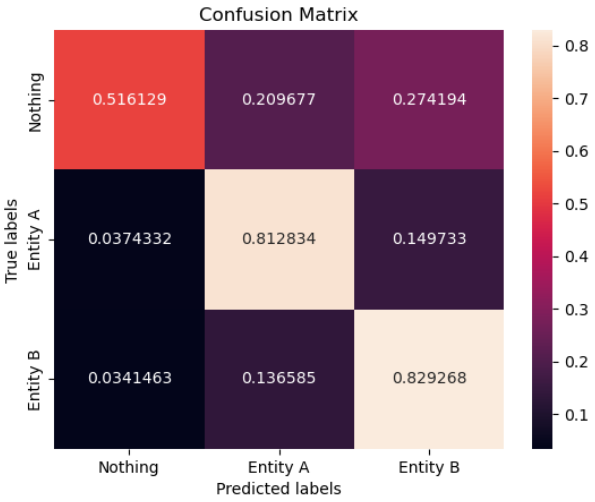

Figure 5: A MLP composed of a single hidden layer and a batch normalization layer produces scores for the three options, then argmax simply returns the label with the highest score. Comparing the predictions with the relative ground truth is possible to compute a confusion matrix. Entities A and B are generally classified correctly with more than 80% of accuracy, however, "Nothing" class is still not achieving satisfying results.

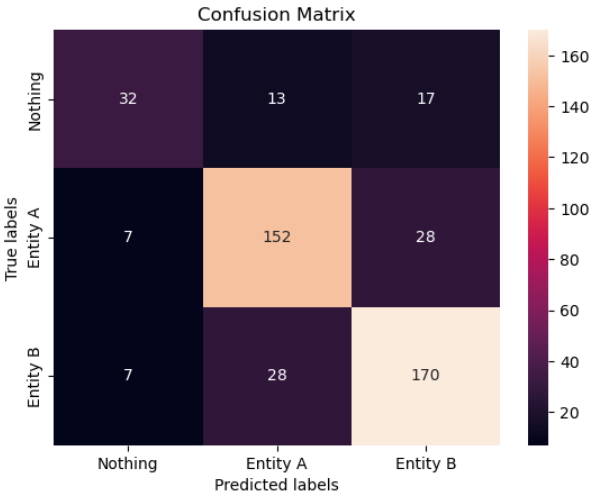

Figure 6: To understand why performance in classifying "Nothing" class is not accurate as it is for the other two classes is possible to have a look at the unnormalized confusion matrix, only 32 over a total of 62 instances are correctly classified for this label, this especially if compared with the results obtained with the others classes it could result not enough accurate, however, we need to consider that for Entity A and B are respectively available 187 and 205 instances, probably a better generalization of this class would have occurred if more training data were available, or maybe defining another strategy for filter output those pronouns where no entity is the referring entity.

References

Sandeep Attree. 2019. Gendered ambiguous pronouns shared task: Boosting model confidence by evidence pooling.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.