Final NLU project

Lorenzo Tomasi (230494)

University of Trento

lorenzo.tomasi-2@studenti.unitn.it

1. Introduction

The objective of the project discussed in this paper is to perform a typical operation in Natural Language Understanding field: joint intent classification and slot filling (or concept tagging), in this case on ATIS and SNIPS datasets. The two different operation can also be performed separately, but doing them together has been demonstrated to be more efficient. The aim was thus to improve the performance of a given baseline, to do that pre-trained model and not, have been used, running them several times to find the best values for the hyperparameters in a Google Colaboratory environment, taking advantage of the GPU available there.

2. Task Formalisation

To better understand the goal of this project, in more detail:

- **Intent classification** is an operation in which given an utterance or a sentence it is aimed to understand its main intention.
- **Slot filling** is a sequence labelling task where the objective is to analyze the words in a sentence to assign a label to those words in the sentence that describe the properties of the request.

As mentioned above the goal was to improve the performance of a given baseline model, specifically by about 2-3%, where these values of the baseline should be:

- ATIS:
  - Slot F1: 92.0%
  - Intent Accuracy: 94.0%
- SNIPS:
  - Slot F1: 80.0%
  - Intent Accuracy: 96.0%

Due to the small dimensions of the two datasets, each model has been trained five times in order to have then an average and the standard deviation of the five runs performances on both datasets. For this reason, first the baseline model values are recalculated in this way, and then the proposed models are analyzed to see the improvements they can achieve.

3. Data Description Analysis

3.1. ATIS

ATIS, (Airline Travel Information System) is a widely used dataset structured with data obtained from the Official Airline Guide (OAG), composed by acoustic speech recordings and their corresponding transcriptions [1]. It has a total of 5871 samples, the version used in this paper had initially train and test dataset only, composed of 4978 and 893 examples respectively. Because of this a validation set had to be created, thus, following what was seen during practical lectures, it has been created a validation dataset with a dimension of 10% of the whole dataset, but dividing the training set only and keeping those intents with only one instance in the training set. As a result, the final dimensions of training, validation and test set are respectively: 4381, 597 and 893. Note, to keep model performance results consistent, the code in which the new version of the dataset was created is separated from the rest of the code and the new datasets were created and saved on Google Colaboratory so that they do not have to be created on every run.

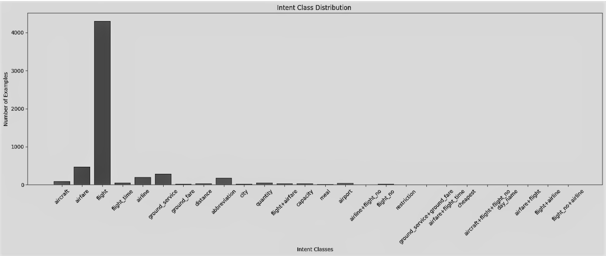

Figure 1: *ATIS intent class distribution*

3.1.1. ATIS statistics

The ATIS dataset has a vocabulary length of 941 words in its entirety, 129 Slots and 26 different intents. From Figure 1 it can be seen as the distribution of the examples on these intents is very unbalanced with more than four thousand with the ”flight” intent.

3.2. SNIPS

SNIPS is another commonly used dataset, composed of user queries or utterances with their corresponding intents and entity labels, in this case the provided dataset was already divided into training, validation and testing set, with sizes of respectively: 13084, 700 and 700 examples for a total of 14484.

3.2.1. SNIPS statistics

The SNIPS dataset has a vocabulary length of 12134 words on the three sets of which it is composed, 72 Slots and 7 different intents. Unlike the ATIS dataset from Figure 2 it can be seen as the examples are more evenly distributed on these intents.

4. Model

4.1. Baseline

ModelIAS: this is a simple model given during the practical lecture, its structure is composed by an embedding layer responsible of mapping the input ids to their corresponding word embeddings, an LSTM (Long Short-Term Memory) layer to encode the word embeddings in a sequence of hidden states taking

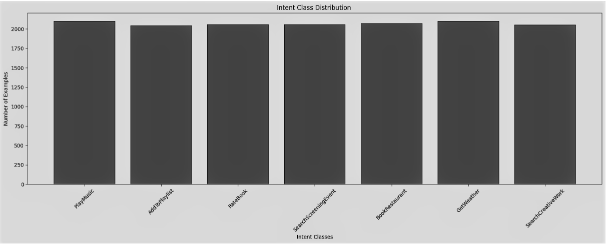

Figure 2: *SNIPS intent class distribution*

the contextual information, two linear layers, one to transforms the hidden states encoder into slot logits and another one that takes the last hidden layer of the encoder and transform it into intent logits. In the code it can also be seen there is a dropout layer that is not used though. Many interesting techniques can be find in this survey on the topic of joint intent detection and slot filling by Weld to improve the values of the baseline[2].

4.2. BiLSTM

Bidirectional LSTM was a simple model obtainable just by modifying this line of the code of the baseline:

```
self.utt_encoder = nn.LSTM(emb_size,
hid_size, n_layer, bidirectional=False)
```

where it can be seen that the bidirectionality of the LSTM encoder was set to false, by setting it to true the LSTM processes the input sequence in forward as before but also in the backward direction. This allow to get contextual information from the previous and the next tokens, clearly improving the understanding. To make the most of this characteristic, the hidden states from the forward and backward direction of the BiLSTM are concatenated, in this way it should be possible to capture more context and thus improve the intent classification, to note, however, that this can lead to an increase in computational load and memory use.

4.3. GRU

After trying the LSTM, an interesting idea was to compare it with GRU (Gated Recurrent Unit), another widely used recurrent neural network architecture in the NLU field, the main difference compared to the LSTM architecture is that GRU is simpler and with less parameters, this usually leads to this usually leads to faster training times and a use of the memory more efficient, this is why it was tried next to the LSTM to see the differences.

4.4. BiLSTM/GRU with CRF

As suggested in the guidelines a BiLSTM (and also the GRU version) with CRF was tried[3]. This model tries to take advantage of a CRF (Conditional Random Field) layer using it from the pytorch-crf library, to use it then the main issue was just to create a mask of the utterances in the preprocessing of the data to exclude the padding positions in the crf calculation.

4.5. BiLSTM/GRU with Focal Loss and Label Smoothing

Looking at the distribution of examples in the intent classes of the ATIS dataset, it can be seen that the distribution is very unbalanced with the bulk of the examples in the ”flight” class, to try to handle this problem two techniques were tried to be implemented, Focal Loss and Label Smoothing, label smoothing is a regularization technique used to improve regularization to increase the uncertainty of the model with respect of a class by a certain parameter that can be set, in this way a smoothed loss is calculated as the sum of the element wise multiplications of the smoothed labels and the probability distribution of the intents[4], while focal loss is a loss that can avoid the overwhelming that the cross entropy loss encounter in dataset with large class imbalance like in the ATIS in this case it is used in an alpha balanced variant like the one proposed in [5], where by increasing alpha more emphasis can be given to examples of less represented classes, while increasing gamma more emphasis is given to hard examples. This whole method added then the issue to find the best values for the three parameters used in the two new functions used.

4.6. BiLSTM/GRU with CRF, Focal Loss and Label Smoothing

This model is simply created to try to take advantages of the CRF for the slot filling and the Focal loss and Label smoothing for the intent, the reasons will be better explained in the Evaluation section.

4.7. BERT

Also following the guidelines a pretrained model was also tried, BERT from Hugging Face[6], a pretrained model that thus has to use its own tokenizer, in this case it was used the bert-base-uncased, for its versatility and for being a ”small” BERT variant that hopefully require less computational time, it required the original sentences as input, so it was reconstructed by joining the words to pass the whole utterance to it, from the tokenized text then the input IDs and an attention mask are obtained and passed to BERT, which outputs can then be used as in the other models passing them in the output layers to obtain the slots and intent.

5. Evaluation (approx. 400-800 words)

5.1. Metrics

5.1.1. Slot F1 Score

Used to evaluate the performances of the slot filling, as required by the guideline it was calculated using the evaluate function in the CONLL file provided (here included at the beginning of the code).

5.1.2. Intent Accuracy

Used to evaluate the performance of the intent classification, calculated using the function ”metrics.classification_report” of sklearn.

Next the results obtained from each model are investigated.

5.2. Baseline

	Slot F1	Intent Acc.
ATIS	92.8% ±0.2	94.2% ±0.2
SNIPS	81.9% ±0.2	96.6% ±0.4

The values are slightly different from those expected, in particular for the values for the SNIPS dataset that are higher by 1.9% and 0.6%. The results of the proposed methods will then be compared to these.

5.3. BiLSTM/GRU

BiLSTM	Slot F1	Intent Acc.
ATIS	94.5% ±0.1	95.5% ±0.2
SNIPS	89.3% ±0.4	97.3% ±0.4

The improvements are with the ATIS dataset 1.7% for the slot F1 and 1.3% for the intent accuracy, while with the SNIPS dataset 7.4% and 0.7% for the slot F1 and intent accuracy respectively. The improvement that stands out immediately is the one of the slot F1 using the SNIPS dataset, probably due to the initial low performance, whereas on the contrary the already high value of the intent accuracy for the SNIPS dataset present a small improvement. For the ATIS dataset the results are promising but not completely satisfactory.

GRU	Slot F1	Intent Acc.
ATIS	94.9% ±0.1	95.9% ±0.4
SNIPS	90.3% ±0.4	97.4% ±0.3

Here we can see as with the use of GRU instead of LSTM the performances of the model are not only comparable but even higher, with improvements of 2.1% for the slot F1 and 1.7% for the intent accuracy when used on the ATIS dataset, and respectively 8.4% and 0.8% for the slot F1 and the intent accuracy when used on SNIPS.

5.4. BiLSTM/GRU with CRF

BiLSTM	Slot F1	Intent Acc.
ATIS	95.0% ±0.1	92.4% ±0.8
SNIPS	91.0% ±0.5	96.8% ±0.1

By adding a CRF layer the expectation was to obtain better slot F1 values, and considering +2.2% on ATIS and +9.1% using SNIPS, it can be said that these expectations have been met. A big disadvantage here appears to be the values of the obtained intent accuracy that compared to the baseline are decreased of 1.8% for the ATIS and with an improvement of only 0.2% on the SNIPS dataset. This can be due of the new way of calculate the total loss during the training loop that probably pose more weight on slot filling features.

GRU	Slot F1	Intent Acc.
ATIS	95.2% ±0.2	93.7% ±0.5
SNIPS	91.5% ±0.4	97.1% ±0.4

As before the performance using GRU are slightly better, but it still has the problem for the intent accuracy, indeed there are improvements of the slot F1 using ATIS of 2.4% and with SNIPS of 9.6% but there is a decrease of 0.5% of the intent accuracy on ATIS and an improvement of 0.5% on the intent accuracy using SNIPS.

5.5. BiLSTM/GRU with Focal Loss and Label Smoothing

BiLSTM	Slot F1	Intent Acc.
ATIS	94.7% ±0.3	96.6% ±0.2
SNIPS	89.8% ±0.7	97.1% ±0.4

Since the greatest difficulties were encountered in improving the intent classification we tried to focus on it with this model by trying an alternative idea. At least for the ATIS dataset it can be said that this idea was successful, while the slot F1 similarly to the BiLSTM model is increased by 1.9%, the intent accuracy has an improvement of 2.4%. However, knowing that this technique was specifically going to work on the issue of unbalance of classes in a dataset one might have expected that the improvement would be significantly less with a very balanced dataset such as SNIPS, the improvements here were indeed of 7.9% for slot F1 score and 0.5% for the intent accuracy.

GRU	Slot F1	Intent Acc.
ATIS	94.9% ±0.2	96.1% ±0.3
SNIPS	90.2% ±0.4	97.4% ±0.2

Using GRU it was interesting to see a small decrease of the improvement on the intent accuracy using ATIS, +1.9% (while the slot F1 has a +2.1%) but a better intent accuracy on SNIPS, +0.8% that still is not as good as it was required though (the slot F1 in this case has an improvement of 8.3%).

5.6. BiLSTM/GRU with CRF, Focal Loss and Label Smoothing

BiLSTM	Slot F1	Intent Acc.
ATIS	95.1% ±0.3	95.3% ±0.3
SNIPS	91.7% ±0.4	97.1% ±0.5

In order to try to obtain the benefits on the F1 slot given by the use of CRF without compromising the values of intent accuracy too much, it was therefore tried to combine the two previous proposals, the results were interesting: +2.3% and +1.1% for slot F1 and intent accuracy respectively using ATIS and +9.8% and +0.5% for slot F1 and intent accuracy respectively using SNIPS. The poor behavior of the model on intent accuracy using ATIS with only the use of CRF is thus effectively mitigated.

GRU	Slot F1	Intent Acc.
ATIS	95.1% ±0.1	94.2 ±0.5
SNIPS	91.6% ±0.2	97.1 ±0.1

Here using GRU the improvements were slightly worse, with generally less improvement and in the case of intent accuracy using ATIS the same value as the baseline.

5.7. BERT

	Slot F1	Intent Acc.
ATIS	92.2 ±0.2	97.6 ±0.2
SNIPS	89.9 ±0.4	98.2 ±0.3

BERT seems to be the best option for intent accuracy, with improvements of 3.4% on its value calculated using ATIS dataset and 1.6% with SNIPS, but with a decrease of 0.6% and an increase of the slot F1 using SNIPS of 8.0% similar to the one of other models tested.

6. Conclusion

In this paper the goal was to improve the performance of a baseline model for joint intent classification and slot filling in terms of slot F1-score and intent accuracy of 2-3%. The main challenge was found in improving the already high intent accuracy that was achieved with the use of the baseline using the SNIPS dataset, a value that only by using the pretrained BERT could be exceeded by more than 1% considering the value obtained here with the baseline. To increase this value different techniques could be considered, a deeper tuning of hyperparameters or models with more complex structures can be considered. Nevertheless, good results were obtained with the models tested, also interesting to see how GRU can be an alternative to LSTM that does not compromise performance, particularly the BiLSTM with the use of focal loss and label smoothing instead of just the cross entropy loss for the intent classification part, to be considered a more in-depth study of its use alongside that of CRF could lead to even more interesting values than those obtained.

7. References

[1] C. T. Hemphill, John J. Godfrey, and G. R. Doddington. ”The ATIS Spoken Language Systems Pilot Corpus,” *In Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania,* June 24-27,1990.

[2] H. Weld, X. Huang, S. Long, J. Poon, and S. Han, ”A survey of joint intent detection and slot-filling models in natural language understanding”, arXiv, cs.CL, 2021.

[3] Z. Huang, W. Xu, and K. Yu, ”Bidirectional LSTM-CRF Models for Sequence Tagging”, arXiv, cs.CL, 2015.

[4] G. Pereyra, G. Tucker, J. Chorowski, Ł. Kaiser, and G. Hinton, ”Regularizing Neural Networks by Penalizing Confident Output Distributions”, arXiv, cs.NE, 2017.

[5] T. Lin, P. Goyal, R. Girshick and K. He, and P. Dollár, ”Focal Loss for Dense Object Detection”, arXiv, cs.CV, 2018.

[6] Q. Chen, Z. Zhuo, and W. Wang, ”BERT for Joint Intent Classification and Slot Filling”, arXiv, cx.CL, 2019 .