HW2 Report

Dario Loi — loi.1940849@studenti.uniroma1.it

Contents

1 Dataset

For this task, we work with a subsampled version of the FEVER dataset (Thorne et al., 2018), which contains additional labels for word-sense disambiguation obtained through AMuSE-WSD (Orlando et al., 2021), as well as Semantic Role Labels, obtained using InVeRo-SRL (Conia et al., 2020).

We have access to a default version of the dataset, split into train, validation, and test sets, as well as an adversarial test set, which is more challenging than the default test set.

2 Augmentation

One of the main tasks of the homework was to produce an augmented version of the dataset, we exploit the additional labels (WSD and SRL information) provided with the dataset together with some extrinsic knowledge of the english language to try and generate new samples while minimizing the risk of introducing noise.

In addition to the provided annotations, we also make use of WordNet (Miller, 1994) to obtain synonym sets for each word in a sentence.

We develop an object-oriented augmentation pipeline that chains a series of transforms in order to generate a set of samples from a single sample, the available transforms are:

- `Synonimization`: This transform replaces a word in the sentence with a synonym of the same word, we use the WSD labels to ensure that the synonym is in the same sense as the original word. This transform can be applied to both the premise and the hypothesis.

- `CopulaContradictor`: This transform replaces the copula in the sentence with its negation, while also negating the hypothesis (`ENTAILMENT` $\leftrightarrow$ `CONTRADICTION`).

- `CopulaInverter`: This transform switches the subject and the object of the copula in the sentence, possibly turning the verb from passive to active form and vice versa, while preserving the logical relationship between the premise and the hypothesis.

- `LengthFilter`: Since the WSD and SRL tokenizers operate slightly different (they mainly handle hyphenation of words differently, one splits along the hyphen and the other doesn't), we use this transform to filter out samples that have a different number of tokens in the premise and the hypothesis. This allows us to easily propagate changes from the WSD tokens to the SRL tokens and back.

Each of these transforms can be composed in a dynamic pipeline[1], and the pipeline can be applied to a dataset to generate new samples. The transforms are parametrized by a given probability, which is used to decide whether to apply the transform to a given sample or not.

`Synonimization` also has the possibility to provide an upper bound on the number of synonyms to consider for each word, this is useful to avoid generating too many samples from a single sample, keeping the dataset's size manageable.

We avoid the use of antonyms and hypernyms/hyponyms, as they tend to introduce noise in the dataset and are more likely to generate samples that are not coherent with the original sample. One clear example of this happening is when transforming adjectives with antonyms, we do not know if this will lead to a `CONTRADICTION` or a `NEUTRAL` relationship between the premise and the hypothesis, and we are therefore unable to apply the correct label transformation automatically (without the use of some LLM model to predict the correct label).

[1] We assume that a `LengthFilter` is required as the first step of any pipeline, else the pipeline will not work as expected.

2.1 Augmentation Parameters and Results

For the generation of the augmented dataset, we build a pipeline as described in table 1, and apply it to a randomly sampled 80% subset of the default training and validation splits, resulting in a roughly $2.5\times$ increase in the number of samples on each.

3 Models

In order to solve the task (Natural Language Inference), we use the `distillRoBERTa` model (Sanh et al., 2019; Lacoste et al., 2019), fine-tuned on the provided dataset, we use a `RoBERTa`-based model since the task is focused on *robustness*, especially on the adversarial examples.

We train two versions of the models, as requested by the assignment's extra task 2. The first one, which we refer to as the *baseline* model, is trained on the original dataset. The second model is trained on a concatenation of the original dataset and the augmented dataset, we refer to this model as the *augmented* model.

We also use the `RoBERTa` tokenizer, which is provided by the `transformers` library, to automatically tokenize the input samples in a way that is conformant with the assignment's specifications.

4 Training and evaluation

We use the `transformers` (Wolf et al., 2020) and `datasets` (Lhoest et al., 2021) libraries to train the models, this results in a simple but quite opaque training process, as the libraries abstract away most of the training loop.

For both models, we use the same hyperparameters, which are listed in table 2.

The training process lasts around 1.5 hours/epoch on a Kaggle P100 GPU.

5 Results

We show the results for both models on both the base set and the adversarial set in tables 3 and 4. Precision, Recall and F1-Score are aggregated over the three classes (`ENTAILMENT`, `NEUTRAL`, `CONTRADICTION`) with weighted averaging, to account for potential class imbalances.

We notice a satisfying performance (around 70% accuracy) on the base test set, while the performance on the adversarial test set is lower, as expected. Interestingly, the augmented model loses a bit of performance on the base test set, but proves to be more robust w.r.t the adversarial test set, with higher accuracy and F1-Score. The augmented model also seems to be more conservative in its predictions, as shown by the higher precision and lower recall.

6 Running the code

In order to allow for easy reproduction of the results, we provide a `requirements.txt` file that lists all the required dependencies, this can be installed with `pip install -r requirements.txt`.

The code is separated into two notebooks (as requested by the assignment), one for the data augmentation and one for the model training. The augmentation notebook produces two dataset files (`augmented_train.jsonl` and `augmented_val.json`) that can be loaded by the training notebook.

The training notebook trains the two models and saves them to disk, it then loads the weights and tokenizers from disk and evaluates the models on the two test sets (expecting that the augmented data is in the same directory as the notebook).

Table 1: Augmentation pipeline for dataset generation.

Transform	Probability	Synonyms	Applies to
`LengthFilter`			Both
`Synonimization`	0.100	2	Premise
`CopulaInverter`	0.500		Hypothesis
`CopulaContradictor`	0.500		Hypothesis
`Synonimization`	0.125	2	Hypothesis

Table 2: Hyperparameters

Parameter	**Value**
`batch_size`	32
`learning_rate`	1e-4
`num_epochs`	2
`warmup_steps`	500
`weight_decay`	1e-3
`seed`	42

Table 3: Results on the base test set

Model	**Accuracy**	**Precision**	**Recall**	**F1**
Baseline	0.704	0.699	0.704	0.694
Augmented	0.683	0.681	0.683	0.672

Table 4: Results on the adversarial test set

Model	**Accuracy**	**Precision**	**Recall**	**F1**
Baseline	0.496	0.511	0.496	0.495
Augmented	0.519	0.542	0.519	0.522

References

Simone Conia, Fabrizio Brignone, Davide Zanfardino, and Roberto Navigli. 2020. InVeRo: Making semantic role labeling accessible with intelligible verbs and roles. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 77–84, Online. Association for Computational Linguistics.

Alexandre Lacoste, Alexandra Luccioni, Victor Schmidt, and Thomas Dandres. 2019. Quantifying the carbon emissions of machine learning. *CoRR*, abs/1910.09700.

Quentin Lhoest, Albert Villanova del Moral, Patrick von Platen, Thomas Wolf, Mario Šaško, Yacine Jernite, Abhishek Thakur, Lewis Tunstall, Suraj Patil, Mariama Drame, Julien Chaumond, Julien Plu, Joe Davison, Simon Brandeis, Victor Sanh, Teven Le Scao, Kevin Canwen Xu, Nicolas Patry, Steven Liu, Angelina McMillan-Major, Philipp Schmid, Sylvain Gugger, Nathan Raw, Sylvain Lesage, Anton Lozhkov, Matthew Carrigan, Théo Matussière, Leandro von Werra, Lysandre Debut, Stas Bekman, and Clément Delangue. 2021. Datasets: A Community Library for Natural Language Processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184. Association for Computational Linguistics.

George A. Miller. 1994. WordNet: A lexical database for English. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Riccardo Orlando, Simone Conia, Fabrizio Brignone, Francesco Cecconi, and Roberto Navigli. 2021. AMuSE-WSD: An all-in-one multilingual system for easy Word Sense Disambiguation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 298–307, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *NAACL-HLT*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Perric Cistac, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. pages 38–45. Association for Computational Linguistics.