Snorkel: a modern Python library in weak supervision

Yue Liu

Introduction to the topic: Weak Supervision

Semi-supervised learning still cannot get rid of the structural assumptions. However, weak supervision can successfully avoid it by using subject matter experts (SMEs). We can achieve weak supervision in three ways: (1) providing higher-level, less precise supervision; (2) giving cheaper, lower-quality supervision; (3) taking existing resources.

Summary of references

New Snorkel(Snorkelv0.9): three important functions

The goal of snorkel is to finish the operations of practitioners in the training data.

Functions	**Techniques**
Labeling Function (LF)	Labeling Training Data
Transformation Functions (TF)	Data Augmentation
Slicing Function (SF)	Monitoring Critical Data Subsets

Unlike the previous version, TF is the most important function because it enables the user to apply the machine learning models.

Two contrast application of Snorkel: Natural Language & Images

Hancock, Braden et al. (2018) introduced the Babble Labble in natural language, a framework with semantic parser converting explanations into executable functions. While Varma Paroma et al. (2017) presented Coral, a paradigm allowing users to write heuristics to label training data to interpret the raw pixels.

Analysis of results including pros and cons. This may include duplicating results by running example SW or testing public SW.

Snorkel will accomplish a huge success in natural language processing, especially in English, because the way to label the data is very similar to the linguists. In Natural Language, Snorkel is still use a lot of time in labelling the data. It needs to be mentioned because the options are very close to what happens in the linguistics field: one use transformational-generative grammar(just like writing the label functions), the other do many tests(just like the option 3 with explanations).

Based on my intuition, I believe the frameworks will have a great progress in basic natural language and images, because some "functions" cannot apply into different languages such as Chinese, and some of the experts in computer science even in language and linguistics might not familiar with a specific language, which will increase the difficulty of solving the problems without explanation. Not all of the experts in CS/LIN would like to change for a very particular case, just like some left-hand side people will never agree to use their right hand, even if they cannot have a better balance or learn to drive.

Let's see two pieces of code to analyze the potential disadvantages. In the analysis of spam, the Snorkel cannot finish the transformation functions without the help of other libraries(i.e., NLTK), especially for the synonyms in adjectives, verbs and nouns. It is also possible that the current library cannot mark the nouns correctly in some languages. For example, in Chinese, zai4jia1(at home) has many different ways to mark. I did test it in different online translators, and I believe that Chinese product might be more helpful in this case.

```
import nltk
from nltk.corpus import wordnet as wn

nltk.download("wordnet")


def get_synonym(word, pos=None):
    """Get synonym for word given its part-of-speech (pos)."""
    synsets = wn.synsets(word, pos=pos)
    # Return None if wordnet has no synsets (synonym sets) for this word and pos.
    if synsets:
        words = [lemma.name() for lemma in synsets[0].lemmas()]
        if words[0].lower() != word.lower():  # Skip if synonym is same as word.
            # Multi word synonyms in wordnet use '_' as a separator e.g. reckon_with. Replace it w
ith space.
            return words[0].replace("_", " ")


def replace_token(spacy_doc, idx, replacement):
    """Replace token in position idx with replacement."""
    return " ".join([spacy_doc[:idx].text, replacement, spacy_doc[1 + idx :].text])
```

In visual relation, I did not find the exact TF function. For me, it means that images are easy to handle by using Snorkel, where we can have a better result by using this library.

Recommendations for a person who want to develop or use such systems

If you want to use the snorkel, please find a mature project or some experts. Just like the Babble Labble, you need more source of supervision in a particular domain. Also, you can have a reference of projects in different programming languages or libraries in Python.

Moreover, you might need to combine it with C++ in the industry. Bach, S. H.et al. (2019) did a case study about Snorkel DryBell in the industry. Snorkel DryBell redesigned the labeling function as a library of templated C++ classes. Just like finance, if you want to be more competitive in the field beyond computer science, you'd better learn C++ first, as well as objective-C.

Conclusions

Snorkel can solve inaccurate supervision and incomplete supervision, except inexact supervision. It is a model based on small-scale data but can achieve the same effects of large-scale. Moreover, Snorkel can create a better prediction model.

Appendix: Helpful Projects for Snorkel;v0.9

Project	**URL**
Hybrid Crowd & Programmatic Labeling	https://www.snorkel.org/use-cases/crowdsourcing-tutorial
Multi-Task Learning	https://www.snorkel.org/use-cases/multitask-tutorial
Recommender Systems	https://www.snorkel.org/use-cases/recsys-tutorial
Information Extraction	https://www.snorkel.org/use-cases/spouse-demo
Visual Relation Detection	https://www.snorkel.org/use-cases/visual-relation-tutorial

References

[1] Bach, S. H., Rodriguez, D., Liu, Y., Luo, C., Shao, H., Xia, C., ... & Kuchhal, R. (2019, June). Snorkel drybell: A case study in deploying weak supervision at industrial scale. In *Proceedings of the 2019 International Conference on Management of Data*(pp. 362-375). ACM.

[2] Hancock, Braden., Liang, Percy., Ré, Chris (2018-05-15). Training with Natural Language [Web blog post]. Retrieved from https://www.snorkel.org/blog/babble.

[3] Ratne, Alex., Bach, Stephen., Varma, Paroma., Ré, Chris (2017-07-16). An Overview of Weak Supervision [Web blog post]. Retrieved from https://www.snorkel.org/blog/weak-supervision .

[4] Ratne, Alex., Bach, Stephen., Ré, Chris (2017-12-01). Programming Training Data [Web blog post]. Retrieved from https://www.snorkel.org/blog/snorkel-programming .

[5] The Snorkel Team., (2019-08-14). Introduction to new Snorkel [Web blog post]. Retrieved from https://www.snorkel.org/blog/hello-world-v-0-9 .

[6] Varma, Paroma., He, Bryan., Ré, Chris (2017-09-14). Snorkel for Image Data [Web blog post]. Retrieved from https://www.snorkel.org/blog/coral.