SoPa++: Leveraging performance and explainability from hybridized RNN, CNN and weighted finite-state neural architectures[†]

Atreya Shankar
atreya.shankar@{uni-potsdam.de,uzh.ch}
Department of Linguistics, University of Potsdam
Department of Computational Linguistics, University of Zurich

November 17, 2020

Abstract

In the last half decade, advancements in Natural Language Processing (NLP) have been predominantly driven by Deep Learning (DL) models. While highly performant, DL models pose a major limitation of being less explainable compared to their classical machine-learning counterparts. This study explores SoPa++; a performant and explainable extension of the hybridized RNN, CNN and weighted finite-state SoPa architecture released by Schwartz et al. (2018).

Contents

[†]SoPa abbreviates **So**ft **Pa**tterns; **++** indicates an expansion or improvement; working title to be fine-tuned with final evaluation

1 Proposal

1.1 Motivation

In the last half decade, advancements in Natural Language Processing (NLP) have been predominantly driven by Deep Learning (DL) models. One of the main advantages of DL models is their ability to function as efficient universal approximators; thereby approximating arbitrarily complex functions and logic programs (Cybenko, 1989; Hornik et al., 1989; Kepner et al., 2018). Besides universal approximation capabilities, cross-validation heuristics help to identify DL models that are generalizable on unseen data sets. As a result, DL models offer decent approximation and generalization capabilities on arbitrary tasks given the appropriate model architecture and hyperparameter configurations.

In recent times and especially in NLP, DL models have been following trajectories of over-parameterization for better approximation and convergence on complex training tasks. Recent studies suggest that the practice of over-parameterization in non-convex optimization environments leads to a combinatorially large number of optimal solutions (Kepner et al., 2018). Given this and our previous perceptions, we postulate that current DL techniques offer the ability to produce *multiple* generalizable models to complex tasks.

$$\overbrace{\text{Approximation} \oplus \text{Generalization}}^{\text{DL models + Cross-validation heuristics}} \Rightarrow \overbrace{\text{Semanticity}}^{\text{Human in the loop}} \tag{1}$$

We argue that the next frontier for DL is the process of selecting semantically relevant models; or otherwise models that actually reflect meaningful considerations which humans could easily pick out and which are not dependent on spurious statistical artifacts. The most direct approach for this could be to involve the human in the loop; whereby the model exposes its internal mechanisms and the human evaluates its relevance and meaningfulness. This idea is reflected above in schematic 1.

While seemingly straightforward, this process of semantic model selection is undermined by the generally unexplainable nature of DL models compared to classical machine learning models such as decision trees (Arrieta et al., 2020). This inherent lack of explainability in DL models further contributes to downstream issues such as adversarial vulnerabilities and ethical conflicts based on data biases (Doran et al., 2017). We therefore make an argument for extensive research into the explainability of DL models.

1.2 Explainable artificial intelligence

Doran et al. (2017) conduct a review of Explainable Artificial Intelligence (XAI) based on corpus-level analyses of NIPS, ACL, COGSCI and ICCV/ECCV papers. Based on their analysis, they characterize AI systems into three mutually exclusive categories of explainability; namely *opaque systems* that offer no insight into

internal model mechanisms, *interpretable systems* where users can mathematically analyze internal model mechanisms and *comprehensible systems* that emit symbols which enable explanations for how conclusions are reached. The study then culminates in the introduction of *truly explainable systems*, where automated reasoning is central to the model's internal mechanisms without significant post-processing for explanation generation.

Arrieta et al. (2020) provide an extensive review of XAI techniques for a spectrum of machine-learning models. We deem this study as very useful because it provides both machine-learning users and practitioners a broad perspective of XAI methodologies. A key part of this study is its focus on post-hoc explainability techniques, which are explainability techniques reserved for models that are not readily interpretable by design. One useful subset of post-hoc explainability in DL would be explanation by simplification, which refers to the process of building a simpler and more explainable model (hereby referred to as a "mimic" model) which resembles the functionalities of its antecedent (hereby referred to as an "oracle" model). It is important to note that explanation by simplification focuses on building *global* mimic models and not *local* mimic models which resemble subsets of the oracle model, as is the case for the Local Interpretable Model-agnostic Explanations algorithm (LIME; Ribeiro et al. 2016).

1.3 Explanation by simplification

Explanation by simplification is generally a difficult task because of natural trade-offs between the oracle model's performance and the mimic model's capacity for explainability (Arrieta et al., 2020). It is therefore vital to limit the loss in oracle model performance while maximizing the capacity for the mimic model's explainability. With this in light, we now look into two explanation by simplification techniques used in sequence learning and NLP.

1.3.1 Simplification of arbitrary oracles

Hou and Zhou (2018) propose a simplification algorithm for producing mimic Finite State Automata (FSA) from arbitrary RNN oracle models. This is done by clustering the hidden states of RNNs based on token-level activations. Strongly clustered hidden states are then grouped together to represent states of a global FSA. A more complex algorithm for producing mimic Weighted Finite State Automata (WFSA) from arbitrary probabilistic oracle models is proposed by Suresh et al. (2019). While learning from arbitrary oracles is seen as advantageous in these studies, we argue that simplification of such arbitrary oracles could lead to mimic models that are not representative of the original oracles. This could negatively impact both the replication of oracle performance and the ability for mimic models to explain the internal mechanisms of their oracles.

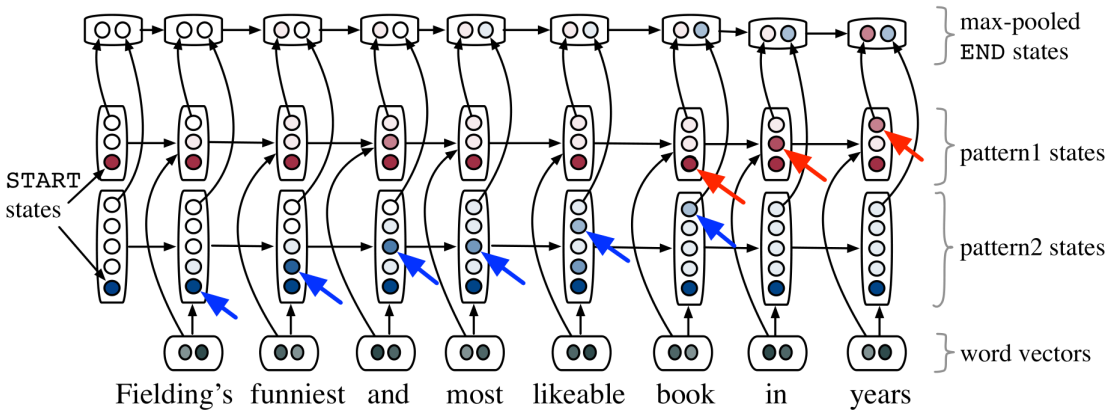


Figure 1: Schematic of the SoPa neural architecture (Schwartz et al., 2018)

1.3.2 Simplification of constrained oracles

To address the aforementioned limitations, we instead look into simplification techniques that use a constrained subset of oracles and mimics which have similar internal mechanisms. In some ways, this can be seen as a process of minimizing a hypothetical "distance" between oracles and mimics. Schwartz et al. (2018) propose a hybridized oracle RNN, CNN and WFSA neural architecture called **So**ft **Pa**tterns (SoPa). SoPa analyzes soft patterns in sequential data and enables simplification of the oracle model into an ensemble of WFSA mimics. Wang and Niepert (2019) propose novel oracle RNN architectures which include discrete centroids that enable smoother simplification into mimic FSA. Similarly, Jiang et al. (2020) provide a novel oracle RNN architecture which allows for smoother simplification into regular expressions.

1.4 SoPa

Based on literature review, source code analyses and email correspondence with all three authors of the studies listed in section 1.3.2; we deem further research into SoPa to be the most feasible given the timeline for this thesis. As SoPa is a complex hybrid neural architecture, we explain its mechanisms briefly and defer the reader to the article for a more in-depth understanding.

1.4.1 Overview

As mentioned previously, SoPa is a hybridized RNN, CNN and WFSA neural architecture whose internal mechanisms are illustrated in Figure 1. SoPa resembles a RNN, CNN and WFSA in the following ways:

RNN: SoPa is a recurrent model which processes tokens one after another. It has a simple recurrent activation function and no external memory.

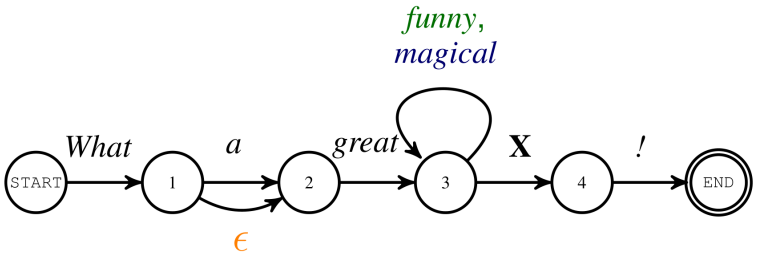


Figure 2: Example mimic FSA simplified from SoPa and its corresponding WFSAs (Schwartz et al., 2018)

CNN: Important hyperparameters in SoPa are the number and length of patterns to learn. Since these patterns have a fixed window size, they resemble extensions of one-layer CNNs.

WFSA: SoPa is designed to resemble a linear-chain WFSA with self-loops and limited ϵ-transitions. These are realized by the construction of a transition matrix, much like the case for learning WFSA directly from input data.

1.4.2 Performance

Schwartz et al. (2018) tested the performance of SoPa on text classification tasks, specifically focusing on binary sentiment polarity detection. SoPa was shown to be competitive against BiLSTM and CNN baselines while using significantly fewer hyperparameters. SoPa was also shown to perform significantly better under small data settings.

1.4.3 Explanation by simplification

The trained SoPa model consists of a transition matrix (or tensor) populated along its diagonals with token and position-specific transition weights. The transition matrix and input pattern hyperparameters can be simplified into textual patterns that fit them best using clustering algorithms. This could be used to identify word-level patterns that were important for the model's classification decisions. These patterns resemble regular expressions with ϵ-transitions and wildcards. A sample pattern for positive sentiment is shown in Figure 2. Schwartz et al. (2018) utilize occlusion to determine which patterns contribute most to positive or negative sentiment.

1.4.4 Limitations

There are several limitations to the current SoPa architecture which were verified through contact with the author(s):

1. SoPa utilizes static word-level token embeddings which might contribute to less dynamic learning and more overfitting towards particular tokens.

2. SoPa encourages minimal learning of wildcards/self-loops and ϵ-transitions, which leads to increased overfitting on rare words such as proper nouns.

3. While SoPa provides an interpretable architecture to learn discrete word-level patterns, it is also utilizes occlusion to determine the importance of various patterns. Occlusion is usually a technique reserved for uninterpretable model architectures and contributes little to global explainability.

4. SoPa was only tested empirically on binary text classification tasks.

1.5 SoPa++

In response to the above limitations, we propose an extension to the SoPa architecture: SoPa++. The exact details of the extension might vary slightly based on time and resources. However a list of open tasks corresponding one-to-one with the limitations listed in section 1.4.4 is provided below:

1. Leverage dynamic sub-word-level embeddings from recent advancements in Transformer-based language modeling.

2. Modify the architecture and hyperparameters to use more wildcards or self-loops, and verify the usefulness of these in the mimic WFSA models.

3. Modify the output multi-layer perceptron layer to a general additive layer, such as a linear regression layer, with various basis functions. This would allow for easier interpretation of the importance of patterns without the use ofocclusion.

4. Test SoPa++ on multi-class text classification tasks, focusing particularly on Natural Language Understanding (NLU) tasks with existing benchmarks. One example is the RASA NLU benchmark (Bocklisch et al., 2017) which contains numerous English language tasks for single-sequence multi-class intent classification.

1.5.1 Research questions

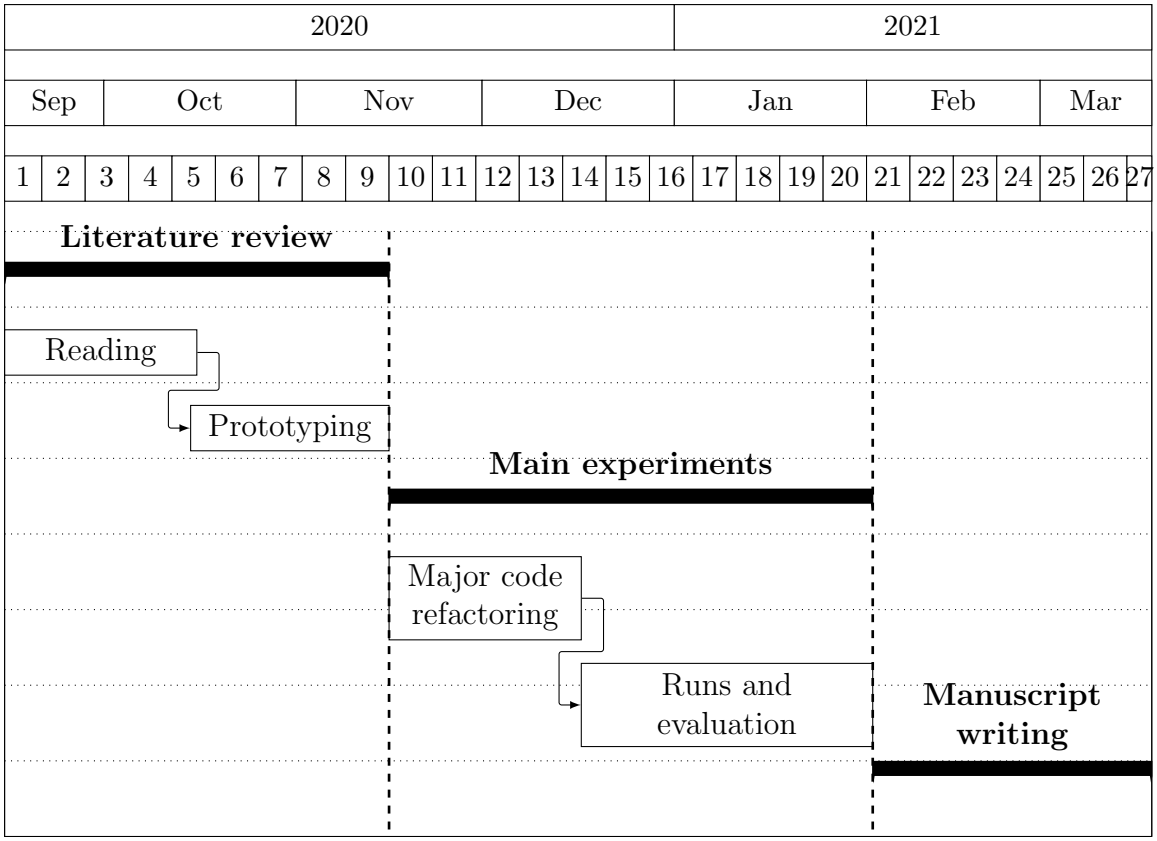
Based on the aforementioned tasks, we formulate three research questions:

1. To what extent does SoPa++ contribute to competitive performance on NLU tasks?

2. To what extent does SoPa++ contribute to improved explainability by simplification?

3. What interesting and relevant explanations does SoPa++ provide on NLU task(s)?

2 Timeline



15.09.20
Internship start

15.11.20
Internal topic agreement

01.02.21
Official topic registration

18.03.21
Manuscript submission ♥

References

A. B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al. Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115, 2020.

T. Bocklisch, J. Faulkner, N. Pawlowski, and A. Nichol. Rasa: Open source language understanding and dialogue management. *arXiv preprint arXiv:1712.05181*, 2017.

G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.

D. Doran, S. Schulz, and T. R. Besold. What does explainable AI really mean? A new conceptualization of perspectives. *CoRR*, abs/1710.00794, 2017. URL `http://arxiv.org/abs/1710.00794`.

K. Hornik, M. Stinchcombe, H. White, et al. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

B. Hou and Z. Zhou. Learning with interpretable structure from RNN. *CoRR*, abs/1810.10708, 2018. URL `http://arxiv.org/abs/1810.10708`.

C. Jiang, Y. Zhao, S. Chu, L. Shen, and K. Tu. Cold-start and interpretability: Turning regular expressions into trainable recurrent neural networks. 2020.

J. Kepner, V. Gadepally, H. Jananthan, L. Milechin, and S. Samsi. Sparse deep neural network exact solutions. In *2018 IEEE High Performance extreme Computing Conference (HPEC)*, pages 1–8. IEEE, 2018.

M. T. Ribeiro, S. Singh, and C. Guestrin. "why should I trust you?": Explaining the predictions of any classifier. *CoRR*, abs/1602.04938, 2016. URL `http://arxiv.org/abs/1602.04938`.

R. Schwartz, S. Thomson, and N. A. Smith. Bridging CNNs, RNNs, and weighted finite-state machines. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 295–305, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1028. URL `https://www.aclweb.org/anthology/P18-1028`.

A. T. Suresh, B. Roark, M. Riley, and V. Schogol. Approximating probabilistic models as weighted finite automata. *CoRR*, abs/1905.08701, 2019. URL `http://arxiv.org/abs/1905.08701`.

C. Wang and M. Niepert. State-regularized recurrent neural networks. volume 97 of *Proceedings of Machine Learning Research*, pages 6596–6606, Long Beach, California, USA, 09–15 Jun 2019. PMLR. URL `http://proceedings.mlr.press/v97/wang19j.html`.