ConceptX: A Framework for Latent Concept Analysis

Firoj Alam[◇] **Fahim Dalvi**[◇] **Nadir Durrani**[◇]

Hassan Sajjad[♣*] **Abdul Rafae Khan**[♠] **Jia Xu**[♠]

{fialam,faimaduddin,ndurrani}@hbku.edu.qa

[◇]**Qatar Computing Research Institute, HBKU Research Complex, Qatar**
[♣]**Faculty of Computer Science, Dalhousie University, Canada**
[♠]**School of Engineering and Science, Steven Institute of Technology, USA**

Abstract

The opacity of deep neural networks remains a challenge in deploying solutions where explanation is as important as precision. We present **ConceptX**, a human-in-the-loop framework for interpreting and annotating latent representational space in pre-trained Language Models (pLMs). We use an unsupervised method to discover concepts learned in these models and enable a graphical interface for humans to generate explanations for the concepts. To facilitate the process, we provide auto-annotations of the concepts (based on traditional linguistic ontologies). Such annotations enable development of a linguistic resource that directly represents latent concepts learned within deep NLP models. These include not just traditional linguistic concepts, but also task-specific or sensitive concepts (words grouped based on gender or religious connotation) that helps the annotators to mark bias in the model. The framework consists of two parts (i) concept discovery[1] and (ii) annotation platform.[2,3]

Introduction

Work done on representation analysis undercover linguistic phenomena that are captured as DNNs are trained towards any NLP task (Belinkov et al. 2017; Liu et al. 2019; Tenney, Das, and Pavlick 2019; Dalvi et al. 2019a). A downside of previous methodologies is that their scope is limited to pre-defined concepts that only reinforce traditional linguistic knowledge and do not reflect on the novel concepts learned by the model, therefore resulting in a narrow view of the knowledge captured by pre-trained Language Models.

We do away with this problem by presenting **a Framework for Latent Concept Analysis (LCA) in deep NLP Models.** Our framework is composed of two modules: i) Concept Discovery and ii) Annotation Platform. We discover latent concepts in deep NLP models using an unsupervised approach (Dalvi et al. 2022) and enable a human-in-the-loop platform to annotate these concepts. The framework is facilitated by auto-labeling the concepts by aligning them to the linguistic ontologies (Sajjad et al. 2022).

Platform Overview

In Figure 1 we present the pipeline of our framework and below we describe these modules:

Concept Discovery

Our method is based on an unsupervised approach to discover latent concepts in the representational space of the pre-trained models (Dalvi et al. 2022). We generate feature vectors (contextualized representation) by doing a forward-pass on the model, and cluster representations using agglomerative hierarchical clustering (Gowda and Krishna 1978) to discover the encoded concepts. Our hypothesis is that contextualized word representations learned within pLMs capture *meaningful* groupings based on a coherent concept such as lexical, syntactic and semantic similarity, or any task or data specific pattern that groups the words together (Sajjad, Durrani, and Dalvi 2021).

Annotation Platform

Once we have discovered the latent concepts captured in the model, we provide a human-in-the-loop framework to analyze and annotate these concepts. To facilitate the effort, we auto-label the concepts with their linguistic connotations wherever applicable.

Annotation Guidelines Our annotation consists of two questions, **Q1:** Does the cluster represent a meaningful concept? **Q2:** Can the neighboring clusters be combined to form a meaningful concept? We provide annotators with specific instructions with examples of what constitutes a meaningful concept based on our definition (Sajjad, Durrani, and Dalvi 2021). The word cluster is represented in a form of a word cloud of examples – See Figure 2), where the relative size of a word in the word cloud depends on the frequency of the word in the data. To understand the context of each word in

[*]This work was carried out while the author was at QCRI.

[1]Code available at: https://github.com/hsajjad/ConceptX

[2]Main platform: https://micromappers.qcri.org/, where project can be created for a new task. Here is an example of the BERT concept annotation task: https://micromappers.qcri.org/project/nx-concept-annotation

[3]https://youtu.be/t8UCP6WPoYg

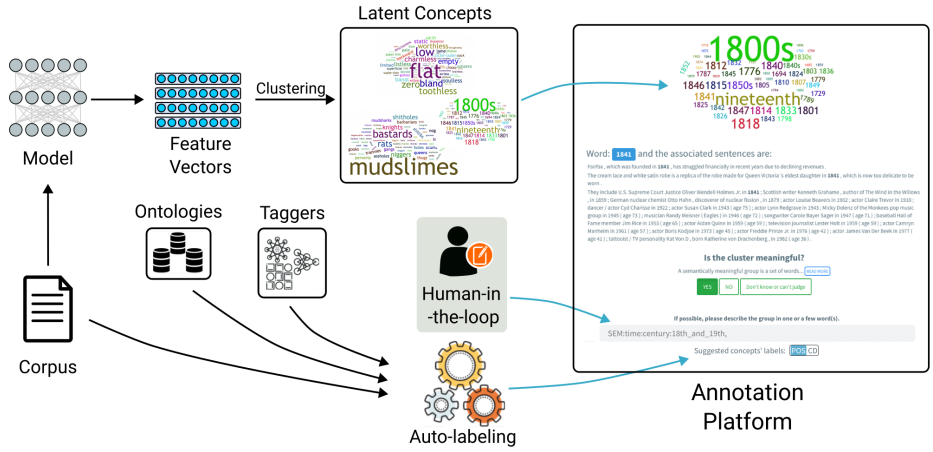

Figure 1: Complete pipeline of the proposed framework for concept analysis in the trained models.

the cluster we also facilitate the annotators with associated sentences in the dataset. More context is enabled through Google search within our framework, where the the annotator can look-up for meaning and more contextual information on the words in the concept. Due to hierarchical clustering, we often expect neighboring clusters to capture very similar concepts that can be conjoined. For example neighboring clusters capturing Hindu and Muslim names can be combined together to form a Hindu-Muslim names concept. Our goal in Q2 is to capture such concepts. Having such an annotation facilitates analysis at a hierarchical level.

Auto-labeling Pre-trained language models have shown to learn rich linguistic concepts (Belinkov et al. 2017; Liu et al. 2019; Tenney, Das, and Pavlick 2019; Durrani et al. 2019). To prevent the human effort of re-annotating such concepts, we integrate an alignment framework (Sajjad et al. 2022) in our platform. We annotated the training data used to obtain latent concepts, with core linguistic concepts (e.g., parts-of-speech, word-net tags etc.) and map the latent concepts to linguistic concepts through an alignment function.

Use Cases

BCN Development

The BERT conceptNet (BCN) dataset[4] development is an example of the utilization of the framework. We had a group of 6 linguists annotate encoded concepts in the BERT-base-cased model. The resulting dataset: BCN is a unique multi-faceted resource consisting of 174 concept labels with a total of 997,195 annotated instances. It can be used by the community to benchmark efforts on interpretability using model's concept instead of relying on extrinsic concepts. The hierarchy present in the concept labels provides flexibility to use data at different granularities.

Task Specific Model Analysis

In addition to the BCN development, we analyzed two transformers – BERT-base-cased (Devlin et al. 2019) and XLM-RoBERT (Conneau et al. 2020) models, fine-tuned towards two tasks: sentiment analysis (SST-2, Socher et al. 2013) and the hate speech detection (HSD, Mathew et al. 2021). Our framework helps visualize how the models segregates negative and positive polarity concepts, in the final layers of the models. See Figure 2 for examples of polarity concepts in the two tasks and (Durrani et al. 2022) for details.

[4]https://neurox.qcri.org/projects/bert-concept-net.html

Figure 2: Polarity Concepts: Negative Sentiment (SST-2), Toxic (right) (HSD)

Related Work

A number of toolkits have been made available to carry analysis of neural network models. Google's What-If tool (Wexler et al. 2019) inspects machine learning models and provides users an insight of the trained model based on the predictions. Seq2Seq-Vis (Strobelt et al. 2018) enables the user to trace back the prediction decisions to the input in NMT models. Captum (Kokhlikyan et al. 2020) provides generic implementations of a number of gradient and perturbation-based attribution algorithms. NeuroX (Dalvi et al. 2019b) and Ecco (Alammar 2021) use probing classifiers to examine the representations in pLMs. Tenney et al. (2020) facilitates debugging of pLMs through intractive visualizations. Our goal is slightly different from these toolkits. Along with the analysis of the latent spaces within pLMs, we intend to annotate latent ontologies learned by these models using human-in-the-loop facilitated by traditional linguistic knowledge. We believe such annotations could be useful to benchmark efforts in interpretation and provide a more realistic view of what these models are capturing.

References

Alammar, J. 2021. Ecco: an open source library for the explainability of transformer language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, 249–257.

Belinkov, Y.; Durrani, N.; Dalvi, F.; Sajjad, H.; and Glass, J. 2017. What do Neural Machine Translation Models Learn about Morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, ACL '17, 861–872. Vancouver, Canada: Association for Computational Linguistics.

Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; and Stoyanov, V. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8440–8451. Association for Computational Linguistics.

Dalvi, F.; Durrani, N.; Sajjad, H.; Belinkov, Y.; Bau, D. A.; and Glass, J. 2019a. What Is One Grain of Sand in the Desert? Analyzing Individual Neurons in Deep NLP Models. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence*, AAAI '19, 6309–6317. Honolulu, Hawaii, USA: AAAI.

Dalvi, F.; Khan, A. R.; Alam, F.; Durrani, N.; Xu, J.; and Sajjad, H. 2022. Discovering Latent Concepts Learned in BERT. In *Proceedings of the Tenth International Conference on Learning Representations*, ICLR '22. Online.

Dalvi, F.; Nortonsmith, A.; Bau, D. A.; Belinkov, Y.; Sajjad, H.; Durrani, N.; and Glass, J. 2019b. NeuroX: A Toolkit for Analyzing Individual Neurons in Neural Networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, AAAI '19, 9851–9852. Honolulu, USA.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT '19, 4171–4186. Minneapolis, Minnesota, USA: Association for Computational Linguistics.

Durrani, N.; Dalvi, F.; Sajjad, H.; Belinkov, Y.; and Nakov, P. 2019. One Size Does Not Fit All: Comparing NMT Representations of Different Granularities. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, ACL '19, 1504–1516. Minneapolis, Minnesota, USA: Association for Computational Linguistics.

Durrani, N.; Sajjad, H.; Dalvi, F.; and Alam, F. 2022. On the Transformation of Latent Space in Fine-Tuned NLP Models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, EMNLP. Abu Dhabi, UAE: Association for Computational Linguistics.

Gowda, K. C.; and Krishna, G. 1978. Agglomerative clustering using the concept of mutual nearest neighbourhood. *Pattern recognition*, 10(2): 105–112.

Kokhlikyan, N.; Miglani, V.; Martin, M.; Wang, E.; Alsallakh, B.; Reynolds, J.; Melnikov, A.; Kliushkina, N.; Araya, C.; Yan, S.; and Reblitz-Richardson, O. 2020. Captum: A unified and generic model interpretability library for PyTorch.

Liu, N. F.; Gardner, M.; Belinkov, Y.; Peters, M. E.; and Smith, N. A. 2019. Linguistic Knowledge and Transferability of Contextual Representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL '19, 1073–1094. Minneapolis, Minnesota, USA: Association for Computational Linguistics.

Mathew, B.; Saha, P.; Yimam, S. M.; Biemann, C.; Goyal, P.; and Mukherjee, A. 2021. Hatexplain: A benchmark dataset for explainable hate speech detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 14867–14875.

Sajjad, H.; Durrani, N.; and Dalvi, F. 2021. Neuron-level Interpretation of Deep NLP Models: A Survey. *CoRR*, abs/2108.13138.

Sajjad, H.; Durrani, N.; Dalvi, F.; Alam, F.; Khan, A. R.; and Xu, J. 2022. Analyzing Encoded Concepts in Transformer Language Models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '22. Seattle, Washington, USA: Association for Computational Linguistics.

Socher, R.; Perelygin, A.; Wu, J.; Chuang, J.; Manning, C. D.; Ng, A.; and Potts, C. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 1631–1642. Seattle, Washington, USA: Association for Computational Linguistics.

Strobelt, H.; Gehrmann, S.; Behrisch, M.; Perer, A.; Pfister, H.; and Rush, A. 2018. Debugging Sequence-to-Sequence Models with Seq2Seq-Vis. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, 368–370. Brussels, Belgium: Association for Computational Linguistics.

Tenney, I.; Das, D.; and Pavlick, E. 2019. BERT Rediscovers the Classical NLP Pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4593–4601. Florence, Italy: Association for Computational Linguistics.

Tenney, I.; Wexler, J.; Bastings, J.; Bolukbasi, T.; Coenen, A.; Gehrmann, S.; Jiang, E.; Pushkarna, M.; Radebaugh, C.; Reif, E.; and Yuan, A. 2020. The Language Interpretability Tool: Extensible, Interactive Visualizations and Analysis for NLP Models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 107–118. Online: Association for Computational Linguistics.

Wexler, J.; Pushkarna, M.; Bolukbasi, T.; Wattenberg, M.; Viegas, F.; and Wilson, J. 2019. The What-If Tool: Interactive Probing of Machine Learning Models. *IEEE Transactions on Visualization and Computer Graphics*, 1–1.