The world around us is constantly changing, and humans are very skilled at adapting to these changes. However, our language tools—virtual assistants like Alexa and Siri, or autocompletion tools like Smart Compose—do not adapt. Since these tools are gradually becoming an important part of how we interface with the digital world, their inability to respond to changes in the physical world poses a major limitation. For instance, if you ask Alexa *what do people do for fun these days*, it responds with answers from the web like *travel* or *spend time with others*, completely disregarding the coronavirus pandemic and the crippling effect it has had on the world. In the best case these limitations lead to humorous errors, but more often than not, they have the ability to cause lasting harm.

Most existing natural language processing (NLP) models, which power our language tools, are not designed to adapt. They are trained to perform a single task which is learned from a fixed dataset, like translating French news articles into English. While performance improvements of such systems on traditional benchmarks have been unprecedented, they tend to fail in vastly different settings. For instance, using a model trained to translate news articles, to then translate medical records which are different in terms of both structure and content, results in poor and unreliable performance. Moreover, most of these systems are black-boxes and it is extremely challenging to understand why they fail.

The goal of my research is to build interpretable NLP models that can adapt to new and changing environments. In the past, I have primarily pursued this goal in two ways:

1. By **analyzing black-box models** to understand how they respond to changes in inputs, making them more interpretable and informing future modeling decisions.

2. By **developing interpretable methods** that allow existing models to adapt to different data distributions by simply memorizing data, without any further training.

1 Analysis of Black-Box Models

Before developing models that adapt, it is crucial to understand how existing NLP systems respond to different inputs and the properties they exhibit while making predictions. Understanding model behavior in this way can shed light on the shortcomings of current models as well as the skills and knowledge that can be exploited for adaptation. In my work, I primarily study language models, which have recently become the back-bone of every state-of-the-art NLP system from question answering to machine translation and text summarization [1, 2]. A language model (LM) defines a probability distribution over strings, assigning higher probabilities to fluent and grammatical sentences and lower probabilities to nonsensical sequences. Here, I present two ways in which I analyzed LMs.

Response to changes in inputs. In my first study, I analyzed how autoregressive language models respond to changes in inputs [3]. Similar to Gmail's Smart Compose which is used to recommend completions to sentences, the autoregressive LM defines a probability distribution over next words, conditioned on the leftward context up to that point. For instance, given the context *I like reading*, is the word *books* more likely to be generated next, or the word *bread*?

Even though these models are used in a variety of NLP systems, we know very little about how they work in practice. Before my study, there were a number of hypotheses regarding how these models use the input context. For instance, it was believed they are effective for very long sequences, and they can use specific information from the distant history to generate the next word. My goal was **to empirically verify these hypotheses by understanding how changing the context in specific ways would affect the LM's ability to generate a sensible next word**. This would show us which aspects or features of the context the LM relies on when making predictions, which in turn can help inform decisions on how to adapt them to vastly different settings.

To investigate these hypotheses, **I designed *ablation* tests that involved removing certain features of the context and measuring how much their absence hurts performance**. For instance, shuffling the words in the context removes the precise ordering of the words but does not change which words appear. Another example is truncating the context to include only the most recent words. This removes features like information that appeared in the distant past, maybe a few sentences ago. I ran a diverse set of such tests on the state-of-the-art recurrent LM of the time, the LSTM. My study not only **made black-box**

LMs more interpretable, it also uncovered surprising facts about how LMs use context. For instance, even though LSTMs were primarily designed to model long sequences, they are only able to process about 200 words on average and while they have a very sharp memory of recent context, the distant history is much fuzzier. Later, the Transformer model was proposed [4] which can look at the distant context more directly, and my study offers a post-hoc explanation for why **Transformer LMs use long-range context more effectively than LSTM LMs**. This opens up the path to a number of exciting research questions about Transformers which I am further investigating.

Linguistic properties exhibited. My collaborators and I then analyzed one of the most widely-used state-of-the-art models, the Transformer-based masked LM called BERT [1]. Given the success of this black-box LM, an important question to consider is whether it uses knowledge about the structure of language to make predictions. Since the model mainly relies on context words as input, this knowledge would have to be acquired implicitly and would be very useful in adapting the model to new settings. But, we know very little about whether these models encode any linguistic structure at all. To investigate this, we study which **linguistic properties the model has acquired as a side-effect of learning to predict missing words** [5].[1] We did this by using the model's internal *attention* mechanisms as classifiers—given a word in a sentence, which other word does the model pay most attention to. Overall, we found that BERT learns **rich linguistic representations** by taking advantage of the syntactic structure of language. This is an important finding because it shows that the model learns core skills for understanding and representing language which can be valuable for adaptation. Our work also contributed to an invited paper to "The Science of Deep Learning" colloquium by the National Academy of Sciences [6].

2 Generalization through Memorization

Most existing NLP models are designed to be effective in a very narrow setting—a single task on a fixed dataset from a single domain, like translating French news articles into English. When such a model is used in a different setting, such as translating medical records, it results in poor and unreliable performance. The typical solution for this is to train a separate model for each domain. For instance, the state-of-the-art English language understanding model BERT [1], has counterparts BioBERT for the medical domain, SciBERT for scientific articles, legalBERT for the legal domain, and many more. However, training separate models and deploying them is expensive, and we would ideally like to train a **single model that is effective in many different settings**. In my research, I have proposed memorizing data in an external datastore [7, 8] in order to adapt a model by transferring skills for a learned task to new data and different domains.

Memorization. The key intuition for memorizing data lies in being able to identify whether two items are similar. For example, you know that the phrases *The author of Pride and Prejudice is* and *Pride and Prejudice is written by* both end in the name of the same author, even if you do not know that *Jane Austen* wrote Pride and Prejudice. So if a model were to memorize the fact that *The author of Pride and Prejudice is Jane Austen*, when met with a new input *Pride and Prejudice is written by*, it could look up the similar example from memory and produce the correct answer. **I show that it is possible for models to *memorize* data by storing the right set of examples in an external datastore and retrieving the most similar ones, the *nearest neighbors*, from it.** Similarity between examples is measured using representations of the context, which are very rich and encode a diverse set of features, as shown in Section 1.

I show that memorization can be used with existing state-of-the-art language models [7] and neural machine translation systems [8], with **no added training costs**. I first show that memorizing data improves performance on fixed datasets which is extremely surprising because these models have been painstakingly optimized for these narrow settings. This demonstrates the effectiveness of memorization as a **core technique to improve model generalization**. I also show that memorization can be used to **scale a model to larger datasets** and to **specialize a model to specific sub-tasks**, by simply **varying the examples stored** in the datastore.

[1] This paper won the Best Paper award at BlackboxNLP 2019.

Domain Adaptation. Making a model effective in domains not seen during training is an extremely challenging problem in NLP. This is because the model must be able to transfer existing skills to a new domain where data comes from a vastly different distribution compared to the data the model was trained on. I have presented **the first adaptation method that can make a single model effective in many domains by simply memorizing domain-specific data, without any in-domain training**. This is achieved by saving domain-specific examples in the datastore and retrieving nearest neighbors from it. For instance, in my work on neural machine translation [8], I show that a single model trained on news articles can be used to effectively translate documents in five new domains, including medical and legal documents, all without any training on data from those domains.

Interpretability. Memorization, as I have proposed, is an extremely effective method to allow existing models to adapt to new data and different domains. Apart from improving model generalization in a variety of settings, it is also highly interpretable. When the model makes a prediction, the **retrieved examples can easily be inspected to understand the model's decision**. For both language models [7] and machine translation systems [8], I have shown precise examples to illustrate why memorization improves generalization. I found that nearest neighbor search is particularly useful for the model to look up rare factual knowledge that needs to be memorized and cannot be inferred, like the fact that *Georges Bizet* is the composer of the opera *Carmen*.

Memorization makes existing models surprisingly effective in adapting to new and changing data distributions. Beyond my prior work, I plan to continue exploring how powerful this approach is in making models more adaptable and interpretable, including via collaborations to improve the guarantees of approximate nearest neighbor search and to improve efficiency through effective use of hardware.

3 Future Research Directions

Humans are highly adept at continually learning and adapting to new tasks and transferring learned skills to unfamiliar settings—we are lifelong learners. NLP systems should similarly be able to adapt to new and changing environments, without resetting every time. My prior work on memorizing data is a promising step in this direction when the model must transfer existing skills to a new setting. In the future, I am excited to develop adaptation methods for NLP models that bring them closer to being lifelong learners, while also making these models more interpretable.

Moving Beyond Single-Task Models. Most existing NLP models are designed to do well on a specific task. While the performance of recent models on existing benchmarks has been unprecedented, they have limited applicability in the real world which is constantly changing. Models that can multi-task, i.e. perform many tasks simultaneously, and those that can build on previously acquired skills and knowledge through sequential learning, are going to be crucial for building useful systems in the future. As the next step, I plan to build models that can multi-task by encoding shared knowledge across different tasks, along with external memories that store task-specific knowledge. My previous collaboration on building models that can multi-task by using single-task models to *teach* the joint model [9], strongly suggests that task-specific knowledge will be crucial for making such models effective. I am also extremely excited to explore the challenges of building NLP models that can learn new tasks sequentially via meta-learning (learning how to learn) combined with task-specific memories that alleviate forgetting.

Model Interpretability and Algorithmic Fairness. Black-box models are pervasive in the field of NLP. However, when résumés are filtered based on gender or a professor is automatically assumed to be male, we need to understand what leads to these decisions and how to remove such biases. Also, understanding how the model operates would allow us to develop robust adaptation strategies, as I have shown in Section 2. For these reasons, exploring methods to make NLP models more interpretable and fair will be a key part of my future research. A question that I am excited to explore next is: to what extent do language models rely on specific examples versus generalizable patterns when making predictions? Is removing specific examples enough to make models less biased? Building on prior work in analyzing how language models use context

[3] and showing the effectiveness of nearest neighbor search using context representations [7], I want to understand what dictates which examples form a new input's nearest neighbors and how this changes as the inputs change. Apart from shedding light on model decisions, this will highlight the model's generalizable knowledge which can be exploited for lifelong learning.

Assessing Progress. Progress in the field of NLP has been incredibly fast-paced in the last few years. In order to avoid drawing spurious conclusions, we must take stock of the claims that are being made and contextualize them with the benchmarks, baselines, experimental design and evaluation metrics used. Moving forward, I plan to expand text generation setups to include evaluation of out-of-distribution performance across a range of metrics. This will not only shed light on the models' ability to adapt but will also encourage the creation of evaluation metrics that are useful in a variety of settings. It is crucial for us, as a community, to constantly re-examine how we conduct empirical evaluations in order to ensure real progress. In the past, I have facilitated such discussions as the co-organizer of the workshop on Methods for Optimizing and Evaluating Neural Language Generation held at NAACL 2019 [10]. In addition, in a recent study with my collaborators on statistical power of NLP experiments, we found that some widely-used NLP datasets are too small to detect whether a new and *improved* model is truly an improvement over the baseline [11]. For these reasons, driving progress through meta-analysis, new benchmarks and new evaluation metrics will be an important part of my future research.

4 Conclusion

As the world around us is constantly changing, we need tools that can adapt to these changes in order to remain useful and relevant, and to avoid causing harm. I believe my goals and plans for the future will help us make progress towards building such NLP systems that can not only adapt, but do so in an interpretable and fair manner.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *North American Association of Computational Linguistics (NAACL)*, 2019.

[2] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[3] **Urvashi Khandelwal**, He He, Peng Qi, and Dan Jurafsky. Sharp nearby, fuzzy far away: How neural language models use context. In *Association of Computational Linguistics (ACL)*, 2018.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[5] Kevin Clark, **Urvashi Khandelwal**, Omer Levy, and Christopher D. Manning. What does bert look at? an analysis of bert's attention. In *BlackboxNLP@ACL*, 2019.

[6] Christopher D. Manning, Kevin Clark, John Hewitt, **Urvashi Khandelwal**, and Omer Levy. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences (PNAS)*, 2020.

[7] **Urvashi Khandelwal**, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations (ICLR)*, 2020.

[8] **Urvashi Khandelwal**, Angela Fan, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Nearest neighbor machine translation. 2020. Under Review.

[9] Kevin Clark, Minh-Thang Luong, **Urvashi Khandelwal**, Christopher D. Manning, and Quoc V. Le. BAM! Born-Again Multi-Task Networks for Natural Language Understanding. In *Association of Computational Linguistics (ACL)*, 2019.

[10] Antoine Bosselut, Asli Celikyilmaz, Marjan Ghazvininejad, Srinivasan Iyer, **Urvashi Khandelwal**, Hannah Rashkin, and Thomas Wolf, editors. *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*. North American Association for Computational Linguistics (NAACL), 2019.

[11] Dallas Card, Peter Henderson, **Urvashi Khandelwal**, Robin Jia, Kyle Mahowald, and Dan Jurafsky. With little power comes great responsibility. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2020.