Investigating AAVE in Question Answering Systems

Taiwei Shi
Advisor: Prof. Diyi Yang
Georgia Institute of Technology
maksimstw@gatech.edu

Abstract

The advancement of technology and social inclusion have encouraged the growth of written texts in dialects. Unfortunately, due to the lack of text corpora, most of the state-of-the-art NLP models are trained by Standard American English (SAE) only. It is important to build NLP technology that is both effective and inclusive. Hence, we investigate the performance of state-of-the-art QA systems on AAVE texts. The performance is examined by converting SQuAD and CoQA to AAVE. Our experiments show that the performance of QA systems degrades significantly when tested on AAVE data in a zero-show setting. While the performance can be partially recovered by incorporating AAVE data in the training set, it still leaves much space for improvement. The dataset is publicly available at: https://github.com/maksimstw/SALT_Dialect_Prompt

1 Introduction

Due to the advancement of technology and social inclusion, underrepresented racial and ethnic groups gain a stronger voice in online communities, literature, and many other fields. Such progress leads to an increase in written texts in many different dialects. Unfortunately, due to the availability of text corpora, most of the state-of-the-art NLP models are trained dominantly by Standard American English (SAE), which makes other dialects, such as African American Vernacular English (AAVE), Indian English (IE), and Singapore English, underrepresented. Table 1 shows common linguistic features present in AAVE as described by Ziems et al. (2022)

One recent development of natural language understanding is question answering. With the growing popularity of virtual assistants, many now seek information and answers from chatbots, which are usually trained on SAE. Figure 1 shows how the presence of AAVE affects the performance of

Passage: *Even before the Norman Conquest of England, the Normans had come into contact with Wales. Edward the Confessor had set up the aforementioned Ralph as earl of Hereford and charged him with defending the Marches and warring with the Welsh. In these original ventures, the Normans failed to make any headway into Wales.*

SAE Q: What country was under the control of Norman barons?
BERT: Wales ✓

AAVE Q: Wat country was under da control of Norman barons?
BERT: the Marches ✗

Figure 1: A passage and question (SAE Q) from SQuAD 2.0, along with their AAVE versions (AAVE Q) and predictions from a BERT model.

BERT (Devlin et al., 2018), a state-of-the-art question answering system, on a question answering dataset, namely SQuAD (Rajpurkar et al., 2018). For example, the original question (SAE Q) is asking about the country that was under the control of Norman barons. Although the AAVE version of the question (AAVE Q) is semantically equivalent to the original question, the presence of lexical and morphosyntactical features confuses the model.

Many have shown that natural language processing tools displayed disparity when the inputs are related to racial and ethnic minorities (Sheng et al., 2019). Groenwold et al. (2020) present intent-similar AAVE/SAE tweet pairs and indicate that AAVE GPT-2 generated segments are more negative than their corresponding SAE segments. Additionally, Nadeem et al. (2020) present a large-scale natural dataset and show that popular models like BERT (Devlin et al., 2018), GPT-2 (Brown et al., 2020), RoBERTa (Liu et al., 2019), and XLNET

Type	Dialect	Example
Lexical	SAE	It was the **right** church.
	AAVE	It was the **rite** church
Negative Concord	SAE	You **don't need any** soap for the clean up.
	AAVE	You **don't need no** soap for the clean up.
Inflection	SAE	Safed is a village that **goes** by numerous other names.
	AAVE	Safed is a village that **go** by numerous other names.
Auxiliaries	SAE	**We are better** than before.
	AAVE	**We better** than before
Have/got	SAE	What law **has** 3 customaries?
	AAVE	What law **got** 3 customaries?
Been/done	SAE	I **have** written it.
	AAVE	I **done** wrote it.
Null genitives	SAE	This is **Martin's** bed.
	AAVE	This is **Martin** bed.
Dey/it	SAE	**There is** some milk in the fridge
	AAVE	**It's** some milk in the fridge
Relative clause	SAE	It's a whole lot of **people who don't** want to go to hell
	AAVE	It's a whole lot of **people don't** want to go to hell

Table 1: SAE/AAVE Comparisons

(Yang et al., 2019) exhibit strong stereotypical bias. Moreover, Xu et al. (2021) discover that detoxification of language models causes a disproportionately large increase in LM perplexity on text with AAVE and minority identity mentions. Moreover, increasing the strength of detoxification amplifies this bias. However, very little research focuses on question answering systems. Furthermore, most of the research fails to comprehensively examine the bias of text generation on dialects other than AAVE. In this paper, we created the AAVE equivalent of SQuAD (Rajpurkar et al., 2018) and CoQA (Reddy et al., 2019) dataset and show that the performance of QA systems degrades significantly when tested on AAVE data in a zero-show setting. While the performance can be partially recovered by incorporating AAVE data in the training set, it still leaves much space for improvement. We argue that the creations of the dataset are vital for improving the understanding of AAVE and developing robust and inclusive NLU models.

2 Dataset

We select SQuAD (Rajpurkar et al., 2018) and CoQA (Reddy et al., 2019) datasets for our experiments on question answering systems. Both dataset contain curated paragraphs and associated questions. The parallel AAVE version of the dataset is obtained by implementing the transformation rules provided by Ziems et al. (2022).

2.1 Source of Questions

We source passages and questions from both SQuAD 2.0 (Rajpurkar et al., 2018) and CoQA 1.0 (Reddy et al., 2019). SQuAD is a reading comprehension dataset, which contains curated paragraphs from Wikipedia and associated questions. SQuAD 2.0 is an extension of SQuAD 1.0 (Rajpurkar et al., 2016) that contains both answerable and unanswerable questions. CoQA (Reddy et al., 2019) is similar to SQuAD. However, while every question in SQuAD is independent, questions in CoQA appear in a conversation and are interconnected.

2.2 Catogories of AAVE Feature

Ziems et al. (2022) systematically catalogue a set of computational rules for inserting AAVE-specific language structures into text. There are two big categories of AAVE features: lexical and morphosyntactical. Lexical features relate to lexicon shift, while morphosyntactic features relate to the grammatical conditions. The examples are given in table 1, and we will now enumerate the ones we implement for SQuAD and CoQA briefly.

Auxiliaries. AAVE allows copula deletion and other auxiliary dropping (Stewart, 2014; Ziems et al., 2022). Hence, we can look for tokens with

AUX part of speech tag to drop.

Completive *done* and remote time *been*. The present time perfect tense can be rendered in AAVE using completive verbal marker *done* and remote time *been* (Green, 2002; Ziems et al., 2022).

Existential *dey/it*. AAVE speakers can indicate something exists by using what is known as an it or *dey* existential construction (Green, 2002; Ziems et al., 2022).

Negative concord. This is the use of two negative morphemes to communicate a single negation (Ziems et al., 2022).

Have/got. The modal and the verb form of *have* can be replaced by *got*, and *have to* can be replaced by *got to* or *gotta* (Trotta and Blyahher, 2011).

Inflection Simple present or past tense verbs might not be inflected in AAVE (Green, 2002).

Null genitives. Any possessive endings such as *'s* are not required to express possession in AAVE (Green, 2002).

Relative clause structures. There is a grammatical option to drop the Wh-pronoun when it is serving as the complementizer to a relative clause (Green, 2002).

Lexical. Some of the most recognizable differences between SAE and AAVE are found in the lexicon and orthographic conventions. Ziems et al. (2022) provides a dictionary that serves as a mapping between plausible synonyms and orthographic variants.

2.3 Dataset Transformation

We implemente the method provided by Ziems et al. (2022) to obtain the AAVE version of SQuAD and CoQA. Ziems et al. (2022) provides a rule-based approach that could transform almost any SAE text to AAVE. Compared to style transfer models (Krishna et al., 2020), the method provided by Ziems et al. (2022) could preserve the meaning of the text and also better capture AAVE morphosyntax. We only converte the questions of the dataset to AAVE, while leaving the passages unchanged. This is because we try to imitate the scenario where an AAVE speaker is trying to use state-of-the-art QA systems. Some samples may be constructed in a way that it is impossible to insert any AAVE feature into the sample. Hence, after the conversion, some samples could still remain unchanged. The estimated percentages of AAVE features in the converted dataset are given in table 2.

Rule	SQuAD	CoQA
Lexical	20.02%	16.70%
Negative Concord	2.48%	0.34%
Inflection	14.33%	6.68%
Auxiliaries	40.12%	42.57%
Have/got	1.41%	0.30%
Been/done	2.26%	1.65%
Null genitives	7.04%	2.56%
Dey/it	0.93%	1.33%
Relative clause	3.70%	0.79%

Table 2: Estimated percentage of AAVE features in the converted SQuAD and CoQA dataset.

3 Experimental Setup

3.1 Model

We use BERT (Devlin et al., 2018) as our QA models for both the SQuAD (Rajpurkar et al., 2018) and CoQA (Reddy et al., 2019) dataset. We finetune BERT for a span selection task. It predicts the probability of every token being the `start` and the `end` of the answer span. Since SQuAD 2.0 introduces unanswerable questions, we treat them as having an answer span with `start` and `end` at the `[CLS]` token.

3.2 Training Settings

We train the BERT model on the entire SQuAD 2.0 and CoQA 1.0 training dataset, including both answerable and unanswerable questions. The evaluation is done in two parts. The first part is done against the entire test set, and the second part is done against the converted questions that contain at least 2 AAVE features. The train/dev/test split of the dataset is done and provided by the SQuAD and CoQA authors [1] [2]. Model configuration and training details can be found in Section A.1

3.3 Evaluation Method

In all of our experiments, we evaluate the QA systems using the official evaluation scripts. Both

[1] https://rajpurkar.github.io/SQuAD-explorer/
[2] https://stanfordnlp.github.io/coqa/

Train	Eval	HasAns-Exact	NoAns-Exact	Overall-Exact
SAE	SAE	73.67	71.32	72.49
	AAVE	55.55 ↓ 18.12	76.18 ↑ 4.86	65.88 ↓ 6.61
	AAVE Morph.	71.41 ↓ 2.26	72.41 ↑ 1.09	71.91 ↓ 0.58
	AAVE Lex.	68.76 ↓ 4.91	73.37 ↑ 2.05	71.07 ↓ 1.42
AAVE	SAE	73.20	73.42	73.31
	AAVE	70.09	72.99	71.54
	AAVE Morph.	72.99	73.83	73.41
	AAVE Lex.	72.44	73.12	72.78

Table 3: SQuAD 2.0 Results. AAVE Morph. means that only morphosyntactical transformation rules are applied to the dataset, while AAVE Lex. means that only lexical transformation rules are applied.

Train	Eval	Exact	F1
SAE	SAE	67.7	77.8
	AAVE	62.5 ↓ 5.2	71.3 ↓ 6.5
	AAVE Morph.	65.6 ↓ 2.1	75.5 ↓ 2.3
	AAVE Lex.	65.1 ↓ 2.6	74.7 ↓ 3.1
AAVE	SAE	67.3	77.4
	AAVE	67.0	76.9
	AAVE Morph.	67.6	77.5
	AAVE Lex.	66.9	76.9

Table 4: CoQA 1.0 Results.

the CoQA and SQuAD official evaluation scripts reports the exact match (EM) and F1 scores over the `HasAns` (answerable) and `NoAns` (non-answerabe) slices.

4 Experiments

We conduct experiments to investigate the following questions: (a) are state-of-the-art LM-based QA models robust and inclusive to the introduction of AAVE features? (b) can we recover the performance by incorporating the AAVE data into the training set?

4.1 Zero-shot Performane

Table 3 and Table 4 shows BERT performance on different variants of SQuAD 2.0 and CoQA. We can see from the tables that BERT performance degrades significantly when testing on AAVE data. When applied both lexical and morphosyntactical transformations, BERT shows a drop of 6.61 overall EM score on SQuAD and a drop of 5.2 EM score on CoQA. This shows that BERT is not robust when the questions contain AAVE features.

4.2 Performance Gap Breakdown

For models trained on SQuAD 2.0, we find that the overall performance drop is largely due to the drop in `HasAns` questions, where for `NoAns` questions, the change in performance is almost negligible or even positive. Upon closer analysis, we can see form Table 5 that there is an increase in the prediction errors for `HasAns` questions being predicted as `NoAns` ones.

We speculate that this is because the AAVE features resemble the unanswerable questions in the training set. Hence, the model tends to be more conservative to the AAVE questions and over-predict `NoAns`. This could also be the reason why there is an increase in performance of the `NoAns` questions.

Original	**HasAns**		**NoAns**
Prediction	**NoAns**	**WrongAns**	**HasAns**
SAE	694	1100	1593
AAVE	**1604**	1180	1417

Table 5: Breakdown of prediction error for the BERT model on SQuAD 2.0. `WrongAns` represents that the model predicts an incorrect span in the context.

In addition, due to the linguistic structures of some questions, 29.39% and 16.19% of the questions in CoQA and SQuAD cannot be converted to AAVE. Hence, to further break down which type of AAVE features cause the most disparity,

Train	Eval	HasAns-Exact	NoAns-Exact	Overall-Exact
SAE (ALL)	SAE (2 fts.)	74.03	75.04	74.53
	AAVE (2 fts.)	51.13 ↓ 22.90	77.50 ↑ 2.46	64.19 ↓ 10.34
	SAE Morph. (2 fts.)	74.03	75.04	74.53
	AAVE Morph. (2 fts.)	65.97 ↓ 8.06	78.33 ↑ 3.29	72.09 ↓ 2.44
	SAE Lex. (2 fts.)	72.18	71.60	71.89
	AAVE Lex. (2 fts.)	63.77 ↓ 8.41	75.19 ↑ 3.59	69.42 ↓ 2.46

Table 6: SQuAD 2.0 Results. Questions that are not converted into AAVE are dropped. SAE/AAVE ALL means that all questions in the dataset are used, even if some cannot be converted into AAVE. SAE/AAVE 2fts means that after conversion, only the questions that contain at least 2 AAVE features are kept in the dataset.

we also evaluate BERT performance on the questions that contain at least 2 AAVE features. We can see from Table 6 that lexical and morphosyntactical features contribute to the drop in performance pretty equally.

4.3 Few-Shot Performance

Next, we want to evaluate whether the disparity can be recovered by incorporating the AAVE data in the training set. To do this, we fine-tune the BERT model on AAVE data for 3 epochs. We can see from Table 3 that BERT performance on SQuAD `HasAns` increases from 55.55 EM score to 70.09. Additionally, we can see from Table 7 that the `HasAns` → `NoAns` type error decreases from 1604 to 864. We can also see similar trend for its performance on CoQA from Table 4, where the F1 score improves from 71.3 to 77.4, which is already quite close to the state-of-the-art performance.

Original	**HasAns**		**NoAns**
Prediction	**NoAns**	**WrongAns**	**HasAns**
SAE	696	1100	1580
AAVE	**864**	1098	1606

Table 7: Breakdown of prediction error on SQuAD 2.0 for the BERT model trained on AAVE.

5 Discussion

By creating the AAVE version of SQuAD and CoQA, we hope to fill in the gap between standard American English (SAE) and other dialects. However, to fully understand and solve the problem, we still need to do the following.

Constructing datasets for spoken problems. Even though there has been an increasing written use of AAVE, question-answering systems and AAVE are also often used in a spoken context. Hence, being a speech phenomenon, a spoken setup would also be important for creating such datasets and evaluating the systems. However, a spoken setup would be quite challenging for data collection. There could be privacy concerns in collecting speech data from the real world. In addition, AAVE and SAE share many linguistic features, which could lead to a relatively low yield for cases containing linguistic features that are unique to AAVE. Moreover, creating a simulated environment itself could be a tedious and challenging task. We hope to address the problems in future work.

Dialects in NLP research. We believe understanding dialects is key to building more inclusive NLP models. We hope our work could raise awareness in the NLP community to devise generalized few-shot or zero-shot approaches to effectively handle AAVE features present in input to NLP models.

6 Conclusion

Through this work, we highlight the need for building more inclusive QA systems for AAVE. To this end, we present the AAVE version of SQuAD and CoQA. Our experiment shows that the state-of-the-art pre-trained language model (BERT) is not robust when tested on AAVE data. We also find that incorporating AAVE data to the training set could partially recover the performance, but it still leaves some space for improvement. We hope our work could shed light on the disparity of language technologies and bring attention to building more inclusive, socially responsible NLP systems.

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda

Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Lisa J. Green. 2002. *African American English: A Linguistic Introduction*. Cambridge University Press.

Sophie Groenwold, Lily Ou, Aesha Parekh, Samhita Honnavalli, Sharon Levy, Diba Mirza, and William Yang Wang. 2020. Investigating african-american vernacular english in transformer-based text generation. *arXiv preprint arXiv:2010.02510*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Moin Nadeem, Anna Bethke, and Siva Reddy. 2020. Stereoset: Measuring stereotypical bias in pretrained language models.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *CoRR*, abs/1806.03822.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Siva Reddy, Danqi Chen, and Christopher D Manning. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2019. The woman worked as a babysitter: On biases in language generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3407–3412, Hong Kong, China. Association for Computational Linguistics.

Ian Stewart. 2014. Now we stronger than ever: African-American English syntax in Twitter. In *Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 31–37, Gothenburg, Sweden. Association for Computational Linguistics.

Joseph Trotta and Oleg Blyahher. 2011. Game done changed: A look at selected aave features in the tv series the wire. *Moderna Sprak*, 105:15–42.

Albert Xu, Eshaan Pathak, Eric Wallace, Suchin Gururangan, Maarten Sap, and Dan Klein. 2021. Detoxifying language models risks marginalizing minority voices.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Caleb Ziems, Jiaao Chen, Camille Harris, Jessica Anderson, and Diyi Yang. 2022. Value: Understanding dialect disparity in nlu.

A Appendices

A.1 Implementation Details

SQuAD We fine-tune over the BERT-base model using the public code [3]. The batch size is set to be 16, and the learning rate of the Adam (Kingma and Ba, 2014) optimizer is set to be 3×10^{-5}. The max sequence length is 384 tokens, and if the context is longer than the maximum sequence length, we take chunks of the up to 128 tokens (`doc_stride = 128`). We fine-tune the model for a total of 3 epochs.

CoQA We fine-tune over the BERT-base model using the public code [4]. The batch size is set to be 12, and the learning rate of the Adam (Kingma and Ba, 2014) optimizer is set to be 3×10^{-5}. The max sequence length is 512 tokens, and if the context is longer than the maximum sequence length, we take chunks of the up to 128 tokens (`doc_stride = 128`). We fine-tune the model for a total of 2 epochs.

[3] https://github.com/huggingface/transformers
[4] https://github.com/adamluo1995/Bert4CoQA