RETHINKING SELECTION BIAS IN LLMS: QUANTIFICATION AND MITIGATION USING EFFICIENT MAJORITY VOTING

Blessed Guda, Lawrence Francis, Gabrial Zencha Ashungafac, Carlee Joe-Wong & Moise Busogi
College of Engineering
Carnegie Mellon University
Pittsburgh, PA 15213, USA
{blessedg,lfrancis,gzenchaa,cjoewong,mbusogi}@andrew.cmu.edu

ABSTRACT

Selection bias in Large Language Models (LLMs) for multiple-choice question (MCQ) answering occurs when models show a preference for specific answer choices based on factors like their position or symbolic representation, rather than their content. This bias can undermine the fairness and reliability of LLM-based systems. In this paper, we first introduce a granular label-free selection bias metric that enables efficient and robust evaluation of selection bias without requiring the answer distributions. Although majority voting, which aggregates predictions across all possible permutations of answer choices, has proven effective in mitigating this bias, its computational cost increases factorially with the number of choices. We then propose Batch Question-Context KV caching (BaQCKV), an efficient majority voting technique, which reduces computational overhead while maintaining the effectiveness of bias mitigation. Our methods provide an efficient solution for addressing selection bias, enhancing fairness, and improving the reliability of LLM-based MCQ answering systems.

1 INTRODUCTION

Selection bias in Large Language Models (LLMs) has been increasingly recognized as a significant challenge, particularly in multiple-choice question (MCQ) answering tasks (Wei et al., 2024b; Zheng et al., 2024; Zong et al., 2023). This bias occurs when models exhibit a preference for certain answer choices based on factors like their position or symbolic representation, rather than the content itself (Wei et al., 2024b). Such bias can distort the reliability and fairness of LLM-based evaluation systems, which are widely used in real-world applications ranging from education to professional testing. Zheng et al. (2024) highlighted the presence of selection bias in LLMs, demonstrating how factors like answer position and symbolic representation can lead to systematic errors in MCQ answering. Several metrics have been proposed to measure the selection bias such as the Choice Kullback-Leibler Divergence (CKLD) (Choi et al., 2024), Standard Deviation of Recalls (RStd) (Zheng et al., 2024), and Relative Standard Deviation (RSD) (Croce et al., 2021; Reif & Schwartz, 2024), which primarily evaluate bias in terms of divergence from ground truth distributions (CKLD) or variability in class-wise performance (RStd and RSD). However, they do not adequately capture the bias exhibited by models to option permutations. Also, the Fluctuation Rate proposed by (Wei et al., 2024a) only considers two permutations of the options, which may not capture the full bias. The primary intuition behind our metric is that *logically, an answer's correctness does not change based on its position in a list of options and we therefore want language models to possess this behaviour*. Addressing the problem of bias requires not just quantifying but also mitigating bias. Prior mitigation strategies like majority voting proposed by Zong et al. (2023) that aggregates predictions across all permutations of answer choices - has been shown to reduce bias, its computational cost increases factorially with the number of choices, making it impractical for large-scale tasks. Thus, a key challenge is to develop an efficient method for bias quantification and bias mitigation that can be integrated into real-world systems. We therefore introduce a **novel bias quantification metric** that evaluates selection bias in LLMs without requiring ground-truth distributions, providing a scalable assessment method. Additionally, we propose **Batch Question-Context KV caching** (BaQCKV), an efficient alternative to majority voting that reduces computational cost while preserving bias mitigation effectiveness.

2 METHODOLOGY

Bias Metric: Reducing bias fundamentally requires that a model maintain its confidence in selecting an option regardless of how the options are permuted. We propose a bias quantification metric that accounts for a model's prediction under all possible option permutations. The input prompt for an MCQ is designed such that each option is assigned an ID that the model selects. For example, these IDs can be letters like A, B, C, or D for an MCQ with 4 options. The permutation of options reorders the options, assigning them to different IDs. The proposed metric computes the variance of the probability assigned to each of the n options under all possible permutations without reference to the ground-truth labels and reports the average of this variance across all options:

$$\text{Bias} = \frac{1}{n}\sum_{i=1}^{n}\text{Var}[\mathbf{P}_i], \quad \text{where} \quad \text{Var}[\mathbf{P}_i] = \frac{1}{m}\sum_{j=1}^{m}(P_{i,j} - \bar{P}_i)^2$$

Here, $\mathbf{P}_i$ is the set of predicted probabilities of option i across all m possible permutations, $P_{i,j}$ represents the probability assigned to option i in permutation j, and $\bar{P}_i$ is the mean probability of option i across permutations.

Majority Voting: A possible mitigation strategy for selection bias is to enumerate the predictions for all possible option permutations and obtain the average prediction for each option. This scheme ensures that an option is always selected with the same confidence regardless of the order in which the options are presented, and can be seen as a majority vote on all options Zong et al. (2023); Guda et al. (2025). Predictions made by the majority vote should have zero bias since all possible option permutations are considered, masking the inherent selection bias of the model and making the majority vote an ideal selection bias mitigation strategy. With the majority voting, the output probability, p_i, for an option i is expressed as;

$$p_i = \frac{1}{m}\sum_{j=1}^{m} p_{ji}$$

where $m = n!$ is the number of possible permutations for n options and p_{ji} is the probability assigned to option i in the jth permutation.

Reducing Computational Costs: The computational complexity of making predictions on all possible permutations is $n!$ for an MCQ, with n options. This can be easily reduced by defining a fixed number k and considering only k permutations instead of $n!$ reduces the computational cost (Guda et al., 2025). Thus, $p_i = \frac{1}{k}\sum_{j=1}^{k} p_{ji}$. However, this scheme can be made even more efficient, without a corresponding loss in bias, by employing a KV cache. To do so, we leverage the insight that while an MCQ consists of a question Q (with or without a context), and a set of options, O, *the question, Q, remains the same across all possible option permutations.*

For k permutations of the options, the original formulation of the majority voting (Guda et al., 2025) requires k passes through the LLM, resulting in an additional overhead of $(k-1) \times |Questions \oplus Context \oplus Options|$ tokens per question ($\oplus$ is a concatenation operation). We however, note that the set of $Questions \oplus Context$ tokens remains constant across all k passes for each question in a batch. To eliminate the redundant computation of these tokens across the batch, we are motivated by the KV cache in (Pope et al., 2023) to introduce the BaQCKV, which caches and reuses the KV states of the $Questions \oplus Context$ tokens for a set of k permutations. This cached KV state is pre-pended to the KV states of the k permuted options. The attention mask of the permuted options is then expanded based on the length of the $Questions \oplus Context$ tokens to ensure that the LLM's attention is correctly computed.

We show in Appendix A.1 that the percentage of tokens by using the BaQCKV is defined by Equation (1).

$$\text{Token savings (\%)} = \frac{(k-1) \times |Q \oplus C|}{k \times |Q \oplus C \oplus O|} \times 100 \tag{1}$$

Algorithm 1 Efficient Majority Inference with BaQCKV

procedure BaQCKVInference($Q_C, O_k, \mathcal{M}$)
  Input: Q_C - Question $\oplus$ Context tokens, O_k - k permutations of options, $\mathcal{M}$ - Language Model, **Output:** $\mathcal{Y}_k$ - Model outputs

  Step 1: Cache Question-Context KV States
  $\texttt{KV}_{Q_C} \leftarrow \mathcal{M}.\text{encode}(Q_C)$
  Step 2: Compute KV States for Permuted Options
  for $i = 1$ to k **do**
    $\texttt{KV}_{O_i}, \texttt{mask}_i \leftarrow \mathcal{M}.\text{encode}(O_i)$
  end for
  Step 3: Merge and Adjust KV States
  for $i = 1$ to k **do**
    $\texttt{KV}_i \leftarrow \texttt{KV}_{Q_C} \oplus \texttt{KV}_{O_i}, \quad \texttt{mask}_i \leftarrow \mathbf{1}_{|Q_C|} \oplus \texttt{mask}_i$
  end for
  Step 4: Compute Batch Outputs
  $\mathcal{Y}_k \leftarrow \{\mathcal{M}.\text{decode}(\texttt{KV}_i, \texttt{mask}_i) \mid i = 1, 2, \ldots, k\}$
  return $\mathcal{Y}_k$
end procedure

In Equation (1), the savings are maximized when $|C|$ is large, as in Retrieval-Augmented Generation (RAG), where redundant computation is minimized. Even when $|C| = 0$, savings persist due to the shared $|Q|$ tokens. Larger permutation sizes k further amplify savings by increasing redundancy in $|Q \oplus C|$ across permutations. Thus, BaQCKV is most effective in tasks with substantial shared context, multiple options, and large permutation sizes.

3 Results

Model Name	**TeleQnA**		**MedMCQA**		**QASC**		**Time Savings (%)**	**Tokens Saved (%)**
	Acc	**Bias**	**Acc**	**Bias**	**Acc**	**Bias**		
Qwen2.5-3B-Instruct	0.801	0.021	0.479	0.058	0.737	0.011	-	-
Qwen2.5-3B-Instruct + MV	0.841	0.000	0.487	0.000	0.947	0.000	50.9%	90.45%
Phi-2	0.760	0.069	0.359	0.082	0.630	0.024	-	-
Phi-2 + MV	0.810	0.000	0.361	0.000	0.940	0.000	64.3%	90.45%
Llama3.2-3B	0.469	0.023	0.370	0.010	0.724	0.005	-	-
Llama3.2-3B + MV	0.656	0.000	0.339	0.000	0.862	0.000	88.6%	90.00%

Table 1: Accuracy and bias values for different models across datasets, along with computational efficiency improvements using Majority Voting (MV).

We evaluated three models (Qwen2.5-3B-Instruct (Bai et al., 2023), Phi-2 (Javaheripi et al., 2023), and Llama3.2-3B(Grattafiori et al., 2024)) models on the TeleQnA (Maatouk et al., 2023), MedMCQA(Pal et al., 2022) and and QASC(Khot et al., 2020) datasets. The results in Table 1 demonstrate the importance of the developed bias metric in effectively quantifying selection bias in LLM-based multiple-choice question (MCQ) answering. The models exhibit varying degrees of bias correlating with the difficulty of the problem, with the highest bias in the MedMCQA benchmark due to its difficulty. This confirms that selection bias is present and measurable using our proposed metric. Notably, after applying majority voting (MV), the bias value consistently drops to 0.00. Additionally, models with majority voting show substantial improvements in accuracy, particularly in QASC, where scores increase significantly (e.g., from 0.630 to 0.940 for Phi-2 and 0.724 to 0.862 for Llama3.2-3B), validating the effectiveness of our metric in capturing and mitigating bias.

Beyond bias reduction, our efficient majority voting method enhances real-world applicability by significantly reducing computational costs. As shown in Table 1, our optimized approach for the majority voting with the KV cache results in token and time savings. The Time Savings metric shows that majority voting with BaQCKV reduces inference time by up to 88.6% (Llama3.2-3B) and 64.3% (Phi-2), while also cutting token usage by over 90% across all models. This efficiency gain is crucial for deploying bias-mitigation strategies at scale, making our approach feasible for real-world applications where computational cost is a limiting factor. In summary, our bias metric provides an effective way to diagnose and measure selection bias, while efficient majority voting ensures that bias mitigation can be applied in practice without excessive resource consumption.

REFERENCES

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report, 2023. URL `https://arxiv.org/abs/2309.16609`.

Hyeong Kyu Choi, Weijie Xu, Chi Xue, Stephanie Eckman, and Chandan K. Reddy. Mitigating selection bias with node pruning and auxiliary options, 2024. URL `https://arxiv.org/abs/2409.18857`.

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: A standardized adversarial robustness benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Blessed Guda, Gabrial Zencha Ashungafac, Lawrence Francis, and Carlee Joe-Wong. Qmos: Enhancing llms for telecommunication with question masked loss and option shuffling, 2025. URL https://arxiv.org/abs/2409.14175.

Mojan Javaheripi, Sébastien Bubeck, Marah Abdin, Jyoti Aneja, Sebastien Bubeck, Caio César Teodoro Mendes, Weizhu Chen, Allie Del Giorno, Ronen Eldan, Sivakanth Gopi, et al. Phi-2: The surprising power of small language models. *Microsoft Research Blog*, 2023.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. Qasc: A dataset for question answering via sentence composition, 2020. URL `https://arxiv.org/abs/1910.11473`.

Ali Maatouk, Fadhel Ayed, Nicola Piovesan, Antonio De Domenico, Merouane Debbah, and Zhi-Quan Luo. Teleqna: A benchmark dataset to assess large language models telecommunications knowledge, 2023. URL `https://arxiv.org/abs/2310.15051`.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. Medmcqa : A large-scale multi-subject multi-choice dataset for medical domain question answering, 2022. URL `https://arxiv.org/abs/2203.14371`.

Reiner Pope, Sholto Douglas, Aakanksha Chowdhery, Jacob Devlin, James Bradbury, Jonathan Heek, Kefan Xiao, Shivani Agrawal, and Jeff Dean. Efficiently scaling transformer inference. *Proceedings of Machine Learning and Systems*, 5:606–624, 2023.

Yuval Reif and Roy Schwartz. Beyond performance: Quantifying and mitigating label bias in llms. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6784–6798, 2024.

Sheng-Lun Wei, Cheng-Kuang Wu, Hen-Hsen Huang, and Hsin-Hsi Chen. Unveiling selection biases: Exploring order and token sensitivity in large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 5598–5621, Bangkok, Thailand, August 2024a. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.333. URL `https://aclanthology.org/2024.findings-acl.333/`.

Sheng-Lun Wei, Cheng-Kuang Wu, Hen-Hsen Huang, and Hsin-Hsi Chen. Unveiling selection biases: Exploring order and token sensitivity in large language models. *arXiv preprint arXiv:2406.03009*, 2024b.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=shr9PXz7T0`.

Yongshuo Zong, Tingyang Yu, Ruchika Chavhan, Bingchen Zhao, and Timothy Hospedales. Fool your (vision and) language model with embarrassingly simple permutations. *arXiv preprint arXiv:2310.01651*, 2023.

A APPENDIX

A.1 PROOF OF TOKEN SAVINGS IN BAQCK

In the original Majority Voting (MV) framework, each question undergoes k passes through the LLM, processing the full sequence of $Q \oplus C \oplus O$ each time. The total token cost per question is:

$$\text{Cost}_{\text{MV}} = k \times |Q \oplus C \oplus O| \tag{2}$$

In BaQCK, the shared $Q \oplus C$ tokens are processed only once, while the O tokens are processed k times. Thus, the total token cost per question is:

$$\text{Cost}_{\text{MV}} = |Q \oplus C| + k \times |O| \tag{3}$$

The token savings is computed as:

$$\begin{aligned}
\text{Savings} &= \text{Cost}_{\text{MV}} - \text{Cost}_{\text{BaQCK}} && (4)\\
&= k \times |Q \oplus C \oplus O| - (|Q \oplus C| + k \times |O|) && (5)\\
&= k \times |Q \oplus C| + k \times |O| - |Q \oplus C| - k \times |O| && (6)\\
&= (k-1) \times |Q \oplus C| && (7)
\end{aligned}$$

Expressing this as a percentage of the original cost:

$$\text{Token savings (\%)} = \frac{(k-1) \times |Q \oplus C|}{k \times |Q \oplus C \oplus O|} \times 100 \tag{8}$$

This result shows that BaQCK significantly reduces token computations, particularly when $|C|$ is large (e.g., in Retrieval-Augmented Generation). Even for small or zero-context cases ($|C| = 0$), savings persist due to shared $|Q|$ tokens. Increasing k further amplifies efficiency by reducing redundant recomputation across shuffled options.