Detecting Expert Users in StackExchange Using Machine Learning

Himani Musku[1], Alea Ritchie[1], Nour Jedidi[2], Rohan Leekha[2], Timothy Reid[2], Courtland VanDam[2]

[1]Illinois Mathematics and Science Academy, [2]MIT Lincoln Laboratory

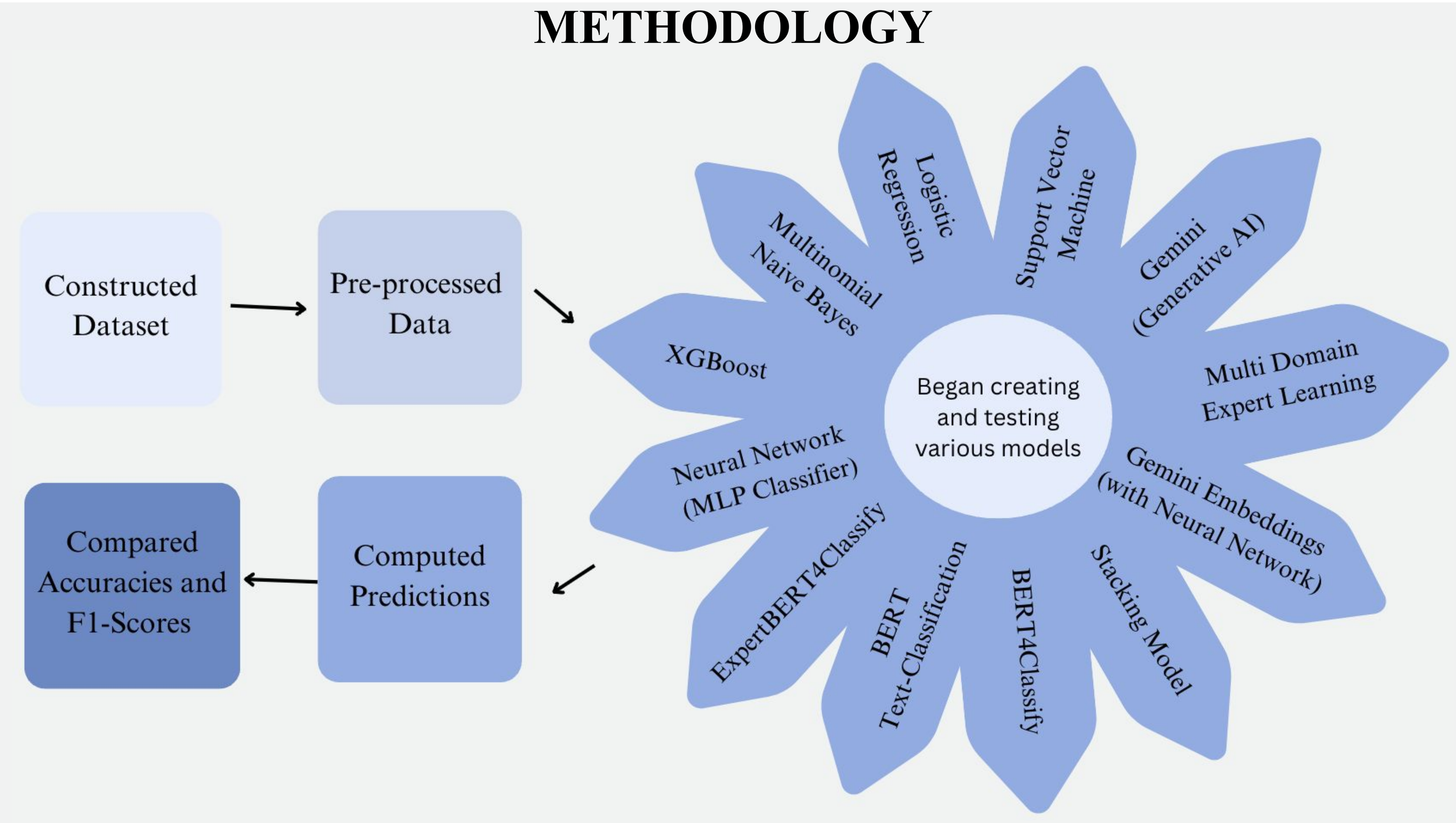
INTRODUCTION

- Online Q&A platforms like StackExchange have grown rapidly in recent years, making it necessary to identify the credibility of users
- Past research has used behavior patterns and user scores to detect expertise (Ban, 2019), while others have focused on textual content alone (Purohit, 2012; Diyanati, 2020)
- Recent work shows that supervised models on Reddit comments can achieve high accuracy in identifying experts (Strukova, 2023)

Our project aims to study:

1. How effectively machine learning models can classify StackExchange users as experts or non-experts using only written text
2. Which classification techniques (among 12 tested) perform best across different evaluation metrics
3. How class imbalance affects model performance and which embeddings are most robust under skewed conditions

METHODOLOGY



We conducted analyses on a dataset consisting of various posts and comments in 12 different forums (e.g: Physics or Biology) written by over 10,000 StackExchange users, who were pre-classified as experts or non-experts based on their StackExchange reputation score.

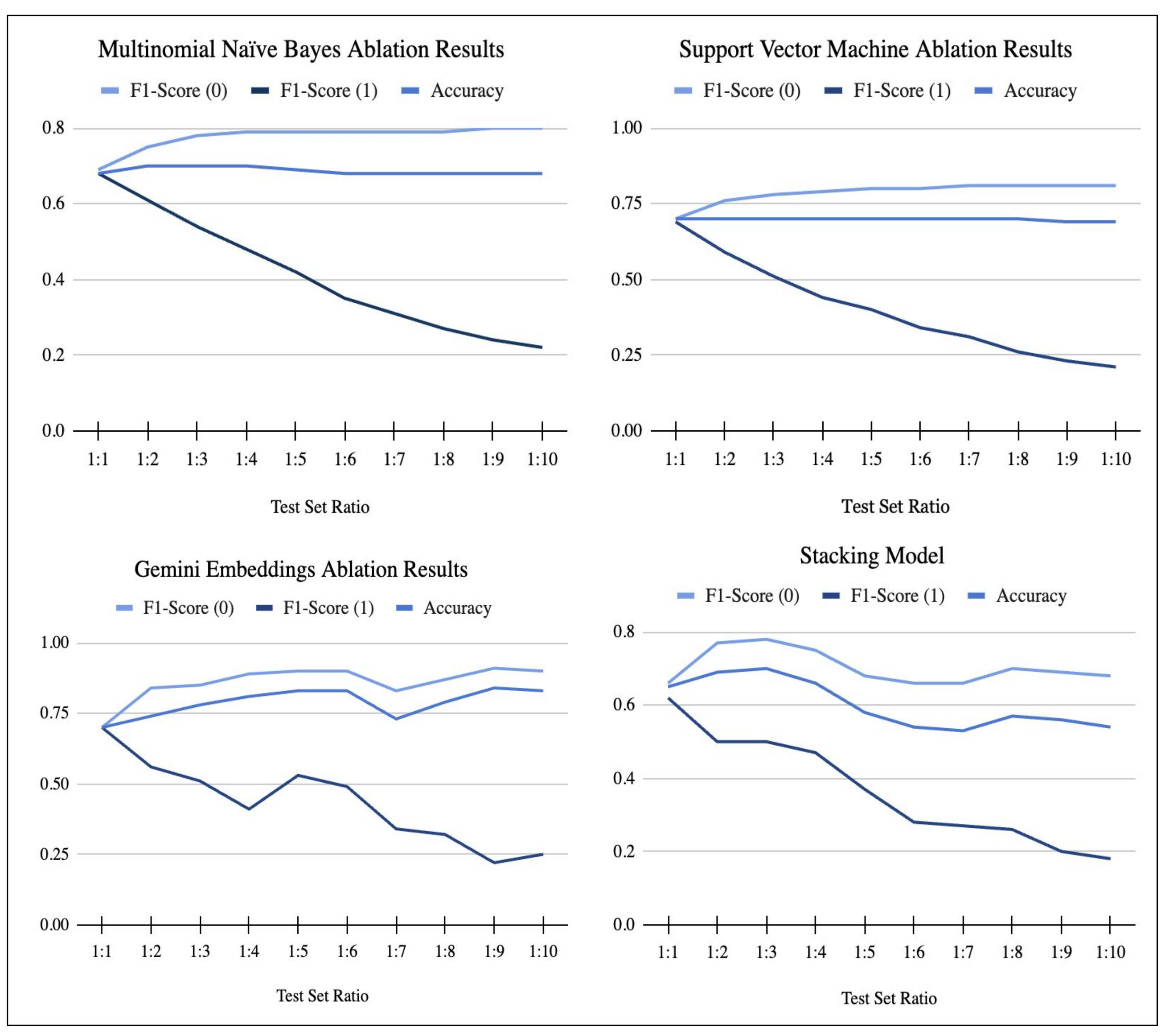


Figure 1. *Ablation Study of Model Performance Under Increasing Class Imbalance.* These plots show the F1 scores for non-expert (0) and expert (1) classes, along with accuracy, across varying test set ratios (1:1 to 1:10) for four different models. Gemini Embeddings demonstrate resistance to increasing imbalance, maintaining higher F1 (1) and accuracy compared to other models.

	Multinomial Naïve Bayes			Support Vector Machine			Gemini Embeddings			Stacking Model		
Test Set Ratio	F1 (0)	F1 (1)	Accuracy	F1 (0)	F1 (1)	Accuracy	F1 (0)	F1 (1)	Accuracy	F1 (0)	F1 (1)	Accuracy
1:1	0.69	0.68	0.68	0.7	0.69	0.7	0.7	0.7	0.7	0.66	0.62	0.65
1:2	0.75	0.61	0.7	0.76	0.59	0.7	0.84	0.56	0.74	0.77	0.5	0.69
1:3	0.78	0.54	0.7	0.78	0.51	0.7	0.85	0.51	0.78	0.78	0.5	0.7
1:4	0.79	0.48	0.7	0.79	0.44	0.7	0.89	0.41	0.81	0.75	0.47	0.66
1:5	0.79	0.42	0.69	0.8	0.4	0.7	0.9	0.53	0.83	0.68	0.37	0.58
1:6	0.79	0.35	0.68	0.8	0.34	0.7	0.9	0.49	0.83	0.66	0.28	0.54
1:7	0.79	0.31	0.68	0.81	0.31	0.7	0.83	0.34	0.73	0.66	0.27	0.53
1:8	0.79	0.27	0.68	0.81	0.26	0.7	0.87	0.32	0.79	0.7	0.26	0.57
1:9	0.8	0.24	0.68	0.81	0.23	0.69	0.91	0.22	0.84	0.69	0.2	0.56
1:10	0.8	0.22	0.68	0.81	0.21	0.69	0.9	0.25	0.83	0.68	0.18	0.54

Figure 2. *Performance of Four Classification Approaches Across Varying Class Imbalance Ratios.* This table compares the F1 scores for expert (1) and non-expert (0) classification and overall accuracy across different class imbalance ratios (from balanced 1:1 to highly imbalanced 1:10) for four models. Gemini Embeddings consistently maintained higher F1 scores for experts, even as the imbalance becomes more extreme, indicating robustness under skewed conditions.

	Run 1 - Gemini Embeddings			Run 2 - Gemini Embeddings			Run 3 - Gemini Embeddings		
Test Set Ratio	F1 (0)	F1 (1)	Accuracy	F1 (0)	F1 (1)	Accuracy	F1 (0)	F1 (1)	Accuracy
1:1	0.7	0.7	0.7	0.69	0.69	0.69	0.73	0.7	0.71
1:2	0.84	0.56	0.74	0.74	0.6	0.68	0.83	0.59	0.76
1:3	0.85	0.51	0.78	0.76	0.53	0.68	0.83	0.59	0.76
1:4	0.89	0.41	0.81	0.88	0.46	0.8	0.86	0.46	0.78
1:5	0.9	0.53	0.83	0.89	0.51	0.82	0.86	0.49	0.78
1:6	0.9	0.49	0.83	0.9	0.47	0.83	0.88	0.39	0.81
1:7	0.83	0.34	0.73	0.75	0.34	0.64	0.89	0.36	0.82
1:8	0.87	0.32	0.79	0.87	0.31	0.78	0.89	0.28	0.81
1:9	0.91	0.22	0.84	0.91	0.3	0.84	0.89	0.33	0.82
1:10	0.9	0.25	0.83	0.9	0.28	0.82	0.91	0.33	0.84

Figure 4: *Consistency of Gemini Embeddings Across Three Independent Runs Under Class Imbalance.* This table presents the F1 scores for expert (1) and non-expert (0) classification, along with accuracy, across 10 test set ratios for all three separate runs using Gemini Embeddings. The results show consistently better performances for experts and stable accuracy across all runs, even as class imbalance increases.

Model	F1-Score (0)	F1-Score (1)	Accuracy
Logistic Regression	0.63	0.68	0.66
Support Vector Machine	0.72	0.68	0.7
Multinomial Naïve Bayes	0.73	0.68	0.7
XGBoost	0.54	0.62	0.59
Neutral Network (MLP Classifier)	0.71	0.71	0.71
BERT Text-Classification	0.66	0.66	0.66
BERT4Classify	0.7	0.65	0.68
ExpertBERT4Classify	0.69	0.64	0.66
Multi Domain Expert Learning	0.69	0.59	0.65
Gemini Embeddings (with Neural Network)	**0.73**	**0.73**	**0.73**
Stacking Model	0.66	0.62	0.65
Gemini (Generative AI)	0.69	0.22	0.56

Figure 3: *Performance Comparison of All Classification Models on Balanced Data (1:1 Ratio).* This table compares 12 models based on F1 scores for non-experts (0) and experts (1), as well as overall accuracy. The Gemini Embeddings with Neural Network model achieved the highest balanced performance, with both F1 scores and accuracy at 0.73.

CONCLUSIONS

- Gemini embeddings performed best for our expert detection task, indicated by its high f1-scores and accuracies
  - Showed resistance to different test set ratios
  - Consistently had accuracies of over 70%
- Future analysis could include:
  - Including users' metadata (e.g. users' voting behavior or whose posts they comment on) along with their written contributions
  - Fine-tuning our models for other platforms or utilizing transfer learning to test their performance on platforms like Reddit or Quora directly

REFERENCES

1. Attiaouia, D., Martin, A., Yaghlane, B. (2017). Belief Measure of Expertise for Experts Detection in Question Answering Communities: case study Stack Overflow. Procedia Computer Science 112, pp. 622–631. https://doi.org/10.1016/j.procs.2017.08.099.
2. Ban, Z., Yan, J., Sun, H. (2019). Identifying Potential Experts on Stack Overflow. In: Sun, Y., Lu, T., Xie, X., Gao, L., Fan, H. (eds) Computer Supported Cooperative Work and Social Computing. ChineseCSCW 2018. Communications in Computer and Information Science, vol 917. Springer, Singapore. https://doi.org/10.1007/978-981-13-3044-5_22.
3. Diyanati, A., Sheykhahmadloo, B. S., Fakhrahmad, S. M., Sadredini, M. H., & Diyanati, M. H. (2020). A proposed approach to determining expertise level of StackOverflow programmers based on mining of user comments. Journal of Computer Languages, 61, 101000. https://doi.org/10.1016/j.cola.2020.101000.
4. Momtazi, S., Naumann F. (2013). Topic modeling for expert finding using latent Dirichlet allocation. WIREs Data Mining and Knowledge Discovery, vol 3, 346-353. https://doi.org/10.1002/widm.1102.
5. Purohit, H., Dow, A., Alonso, O., Duan, L., & Haas, K. (2012). User taglines: Alternative presentations of expertise and interest in social media. International Conference on Social Informatics. https://doi.org/10.1109/SocialInformatics.2012.68.
6. Strukova, S., Ruipérez-Valiente, J. A., & Gómez-Mármol, F. (2023). Computational approaches to detect experts in distributed online communities: A case study on Reddit. Cluster Computing, 27, 2181–2201. https://doi.org/10.1007/s10586-023-04076-w.

This material is based upon work supported by the Department of the Air Force under Air Force Contract No. FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of the Air Force.