A Framework for Extracting ICD-10 Codes from Visit Dialogues with a System One Model

Abhinav Srivastava
Independent researcher
sabhinav@bu.edu

***Abstract*—We assign one ICD-10-CM code to a doctor-patient dialogue with TypeSafe Jev, a *System One* model. The model writes no text. It selects from a list of options and gives a probability for each option. The model walks down the code tree and keeps the 5 best paths. Local code adds candidate codes and uses no model request. The model then selects one code from the candidate list. The method uses no task training, and no labeled example is in the model input. On 996 held-out MedSynth dialogues, the method gives the exact code for 62.0% of the dialogues (±3.0) from the 2,032 benchmark codes, and for 57.3% from all 74,719 codes. A trained text classifier gives 34.3%. To our knowledge, these are the first ICD-10-CM results on MedSynth dialogues. The code, the row lists, and one result record for each dialogue accompany this paper.**

I. Introduction

Published systems for automated ICD coding read a note or a discharge summary, and most of them train on MIMIC data, which needs credentialed access [1]–[3]. Ambient documentation tools start from the dialogue. We found no published code-level ICD-10-CM result from dialogues over more than 1,000 codes. Text-writing models also write codes that do not exist or do not match: GPT-4 gives the correct code for 33.9% of official code descriptions [4].

A *System One* model has a different interface [5], [6]. It takes an input (the *state*) and typed questions. A *Choice* question has up to 255 options, and the model returns one probability for each option [7]. The model cannot give a code that is not in the list. We use Jev (`jev-1.13.0`). We found no result for this model on a medical task.

We test on MedSynth [8], a public synthetic benchmark. GPT-4o agents wrote a scenario from an ICD-10 description, then a note, then a dialogue. It has 10,213 usable dialogues with one ICD-10-CM label each, over 2,032 codes. We use the dialogues only. One other work predicts ICD-10 codes from the MedSynth *notes* with trained BERT models [9]. That work and this paper do not measure the same task.

II. Method

Fig. 1 shows the method, and Fig. 2 shows one dialogue.

Walk. The first Choice question is over the code groups of ICD-10-CM (for example M20–M25). We leave out the external-cause groups [10]. Each later request holds one Choice question for each open path, and the options are the children of the current code. After each level the method keeps the 5 paths with the best geometric mean of step probabilities [11]. An early wrong pick is final when a method keeps one path [12].

Local candidates. Three steps add candidate codes. They run locally and use no model request. *Named codes*: at most 3 codes whose title, without the words "other" and "unspecified", occurs as a phrase in the dialogue. *Retrieved codes*: a sentence-vector model (`bge-base-en-v1.5` [13]) scores each code title against each turn of the dialogue, and the 20 best codes become candidates. *Relatives*: all codes in the 3-character categories of the 5 kept codes. We also tested a late-interaction model [14] for retrieval. On 495 development dialogues it put the correct code in its 10 best for 36.4%. The sentence vectors gave 56.8%, so we did not use it.

Final pick. One Choice question asks which candidate is the primary diagnosis. The list has 34 codes on average with the benchmark codes and 72 with all codes. The options show code titles and synonyms, and the option keys are neutral (`c0`, `c1`, ...), so the model sees no code strings. On development rows this change corrected 8 dialogues and lost 0. The probability of the selected option is the confidence of the answer.

Two code sets. With *all codes*, the method selects from 74,719 codes. With *benchmark codes*, it selects from the 2,032 codes that occur as MedSynth labels. This code set uses the list of labels and no labeled example. The list comes from all 10,213 rows, test rows included. The rows outside the test set give the same list. Each trained classifier has such a list by construction.

III. Setup

Rows. The development set has 495 labeled dialogues, and we made the method choices on its first 300. We drew 1,000 other dialogues with a fixed seed for the test set. We removed 4 of them because their text also occurs in a development row, so 996 remain. Each reported setup ran one time on the test set. An earlier setup (the walk with named codes) ran on the test set first. We then built the retrieval step, the relatives step, and the neutral keys. We saw its aggregate score and no test row.

Label words. The MedSynth prompts tell the writer model to use the key words of the code description. The exact label description occurs in 16.0% of all MedSynth dialogues and in 171 of the 996 test dialogues. We report results for the other rows ("not spoken") in a separate column.

Baselines with no model request, all over the benchmark codes. *Title match*: the first named code. *Vector search*: the best retrieved code. *Trained*: a linear support vector classifier on TF-IDF features. We trained it on 9,186 MedSynth dialogues

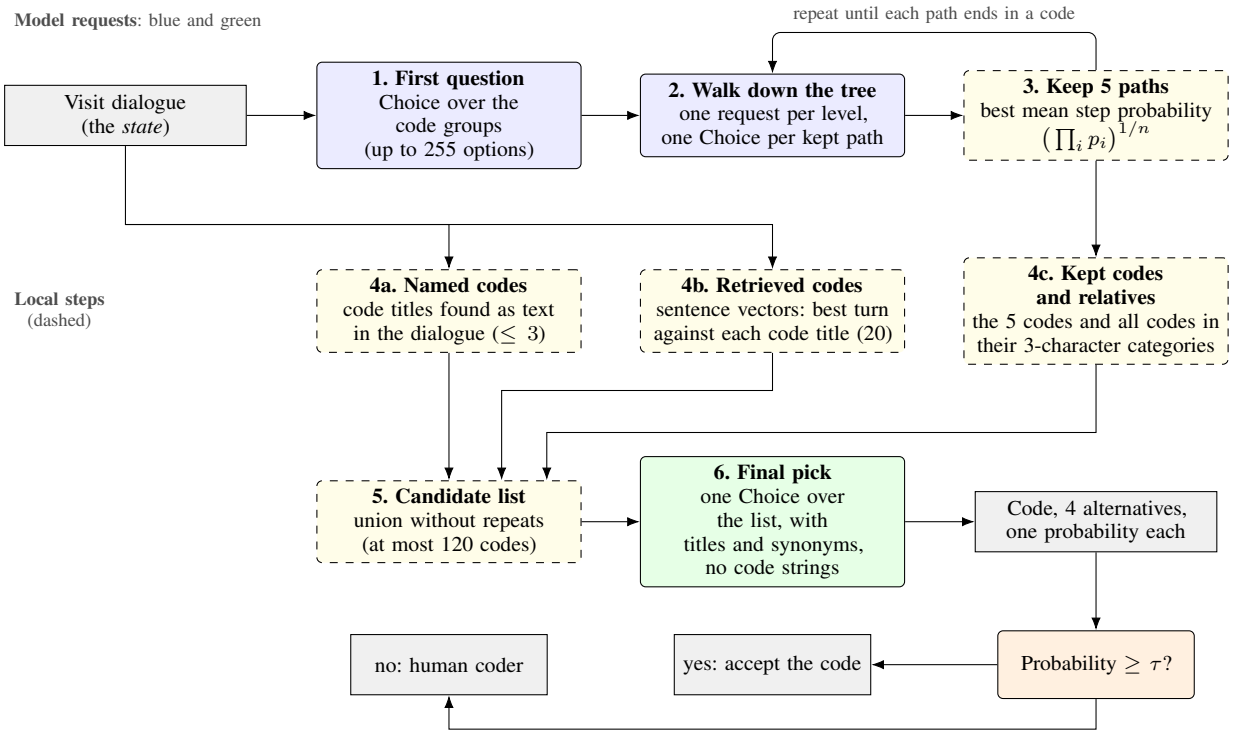

Fig. 1. The method. Blue and green boxes are model requests (4.9 for each dialogue on average with the benchmark codes, 5.3 with all codes). Boxes with a dashed border run locally. Grey boxes are inputs and outputs. The orange box compares the final probability with a threshold τ that the user sets.

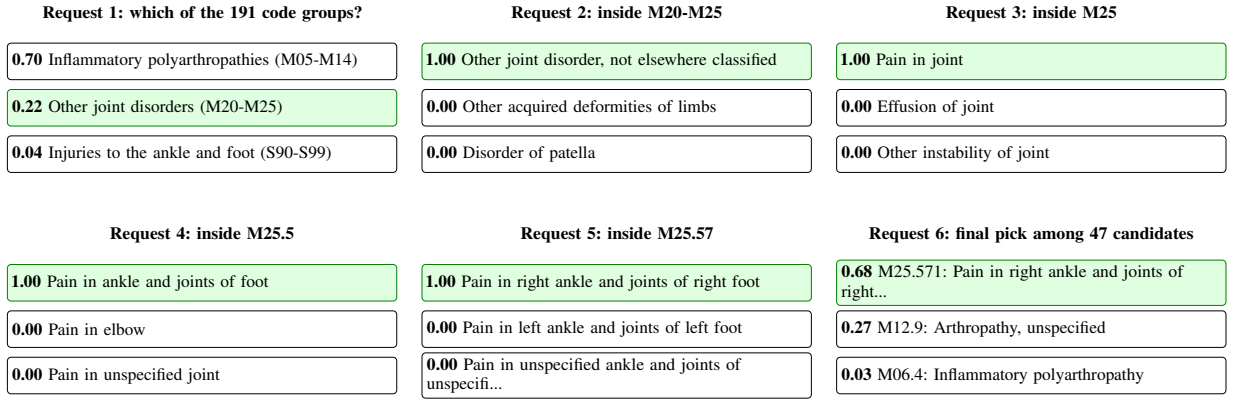

Fig. 2. All 6 model requests of the full method with the benchmark codes for development dialogue 898 (label: Pain in right ankle and joints of right foot). Green marks the path to the final answer. Each panel shows the question on that path, with its 3 best options. In request 1 the correct group is second (0.22). The method keeps 5 paths, so the correct path stays alive. The final pick sees titles and no code strings. The figure adds the code strings for the reader.

(4.5 for each code). These dialogues are not in the test set, and their text does not occur in a test dialogue.

Earlier versions of our method, over all codes. *One path*: a walk with 1 kept path and no final pick. *First method*: a walk with 5 kept paths and a final pick over those 5 codes.

IV. Results

Table I gives the test results. The full method gives the exact code for 62.0% of the dialogues with the benchmark codes and for 57.3% with all codes. With all codes, the full method corrected 64 dialogues of the first method and lost 25 ($p < 0.001$, exact sign test). No test row is missing.

The final pick is the weakest step. With the benchmark codes, the label is in the candidate list for 86.4% of the dialogues, and the first answer is exact for 62.0%. In 30 of the 107 wrong development answers, the answer has the correct category and differs from the label in the word "other" or "unspecified".

The probability as a gate. A tool can accept an answer above a threshold and send the other dialogues to a human coder [15]. With the benchmark codes and a threshold of 0.9, the method accepts 59.2% of the dialogues, and 79.3% of those answers are exact (Fig. 3). The area under the ROC curve is

TABLE I

996 TEST DIALOGUES (PERCENT). NOT SPOKEN: THE LABEL DESCRIPTION DOES NOT OCCUR IN THE DIALOGUE (825 ROWS). FIRST 3: THE 3-CHARACTER CATEGORY IS CORRECT. IN LIST: THE LABEL IS A CANDIDATE. ONLY THE TRAINED BASELINE USES LABELED EXAMPLES.

Setup	Exact	±	Not spoken	First 3	In list
Benchmark codes (2,032)					
Title match	16.8	2.3	6.2	20.1	19.6
Vector search	25.2	2.7	21.7	30.5	66.9
Trained	34.3	2.9	28.8	62.6	–
Full method	**62.0**	3.0	56.1	72.2	86.4
All codes (74,719)					
One path	43.5	3.1	39.0	58.3	43.5
First method	53.4	3.1	48.5	66.7	65.4
Full method	**57.3**	3.1	51.5	71.7	78.7

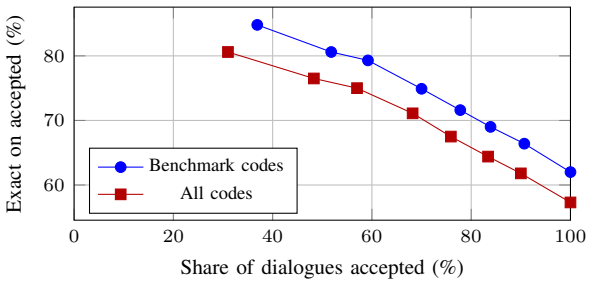

Fig. 3. Full method, test dialogues. Each point is one threshold on the final probability (0 to 0.99, from right to left).

0.77. The probability is too high as a number: its mean is 84.3%, and the expected calibration error [16] is 22.3 points. A user must set the threshold on labeled data.

V. LIMITS

- MedSynth is synthetic, and its dialogues repeat label words. In 518 test rows, not all content words of the label occur in the dialogue. On these rows exact accuracy is 43.1% (benchmark codes) and 37.6% (all codes).
- An LLM assistant reviewed the 93 wrong answers of the first method on 200 development dialogues in one pass. It found 18 labels that the dialogue does not support. No certified coder checked this review.
- The task has one code for each dialogue. Real visits have several codes.
- We saw the aggregate test score of one earlier setup before we built the last three steps.
- We ran no text-writing model on the same rows, and we tested one model version.
- MedSynth is public, and the vendor does not describe the training data of the model. We cannot exclude that the model saw these dialogues and their labels.

VI. RELATED WORK

Supervised coders assign many codes to a discharge summary and train on MIMIC data [1]–[3]. Their multi-label F1 values cannot be compared with single-label accuracy. Boyle et al. [17] walk the ICD-10 tree with GPT-4 and get no probability for a code. Kwan [18] adds retrieval before a pick step. Yuan et al. [19] report that a pick step is more accurate when the options show descriptions and no code strings. Our neutral-key result on development rows agrees with this. Earlier work on a diagnosis from dialogue uses small label sets [20], [21].

VII. FUTURE WORK AND CONCLUSION

This framework remains to be tested on better benchmarks: de-identified visit dialogues, labels from certified coders, and several codes for each visit. A comparison with text-writing models on the same rows is also open.

A model that only selects options and gives probabilities codes a MedSynth dialogue to the exact ICD-10-CM code for 62.0% of the dialogues from 2,032 codes, with no task training. A trained text classifier on the same data gives 34.3%. With all codes, the full method gives 57.3%, against 53.4% for the first method and 43.5% for one path. The final pick is the step to improve next. The long version of this paper (`paper_full.pdf` in the repository) has all development runs, more measures, and the full error review.

REFERENCES

[1] J. Mullenbach, S. Wiegreffe, J. Duke, J. Sun, and J. Eisenstein, "Explainable Prediction of Medical Codes from Clinical Text," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 1101–1111.

[2] C.-W. Huang, S.-C. Tsai, and Y.-N. Chen, "PLM-ICD: Automatic ICD Coding with Pretrained Language Models," in *Proceedings of the 4th Clinical Natural Language Processing Workshop*, 2022, pp. 10–20.

[3] J. Edin, A. Junge, J. D. Havtorn, L. Borgholt, M. Maistro, T. Ruotsalo, and L. Maaløe, "Automated Medical Coding on MIMIC-III and MIMIC-IV: A Critical Review and Replicability Study," in *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2023, pp. 2572–2582.

[4] A. Soroush, B. S. Glicksberg, E. Zimlichman, Y. Barash, R. Freeman, A. W. Charney, G. N. Nadkarni, and E. Klang, "Large Language Models Are Poor Medical Coders — Benchmarking of Medical Code Querying," *NEJM AI*, vol. 1, no. 5, p. AIdbp2300040, Apr. 2024.

[5] TypeSafe AI, "AI primer," TypeSafe documentation, 2026, accessed 2026-09-19. [Online]. Available: https://docs.typesafe.ai/introduction/machine-learning-primer

[6] ——, "Models," TypeSafe documentation, 2026, accessed 2026-09-19. [Online]. Available: https://docs.typesafe.ai/models

[7] ——, "Choice," TypeSafe documentation, 2026, accessed 2026-09-19. [Online]. Available: https://docs.typesafe.ai/primitives/choice

[8] A. Rezaie Mianroodi, A. Rezaie, N. G. Todorov, N. A. Friedrich, M. P. Mogollon, A. Hernandez-Tirado, G. Lopez Garcia, C. Rakovski, and F. Rudzicz, "MedSynth: Realistic, synthetic medical dialogue-note pairs," arXiv preprint arXiv:2508.01401, 2025. [Online]. Available: https://arxiv.org/abs/2508.01401

[9] J. Roche, "Description-level leakage in synthetic clinical benchmarks: a cautionary tale from ICD-10 coding," GitHub repository and Quarto report, 2026, in preparation, not peer reviewed. Title from the repository README. The Quarto report in the repository has the title "Notes to ICD-10: Automated Clinical Code Prediction from Free-Text Notes Using Hierarchical Deep Learning". Commit bda428c (2026-07-07). MIT license. Accessed 2026-09-19. [Online]. Available: https://github.com/Sidney-Bishop/notes-to-icd10

[10] Centers for Medicare and Medicaid Services and National Center for Health Statistics, "ICD-10-CM Official Guidelines for Coding and Reporting, FY 2026," https://www.cdc.gov/nchs/icd/icd-10-cm/files.html, 2025, section I.C.20: an external cause code can never be a principal (first-listed) diagnosis. Accessed 2026-09-19.

[11] TypeSafe AI, "Hierarchical classification," TypeSafe documentation cookbook, 2026, numbers from jev-1.12. Accessed 2026-09-19. [Online]. Available: https://docs.typesafe.ai/cookbooks/hierarchical_classification

[12] Y. Prabhu, A. Kag, S. Harsola, R. Agrawal, and M. Varma, "Parabel: Partitioned Label Trees for Extreme Classification with Application to Dynamic Search Advertising," in *Proceedings of the 2018 World Wide Web Conference (WWW '18)*, 2018, pp. 993–1002.

[13] S. Xiao, Z. Liu, P. Zhang, N. Muennighoff, D. Lian, and J.-Y. Nie, "C-Pack: Packed resources for general Chinese embeddings," in *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2024, pp. 641–649, arXiv:2309.07597.

[14] O. Khattab and M. Zaharia, "ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 39–48.

[15] Y. Geifman and R. El-Yaniv, "Selective Classification for Deep Neural Networks," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, arXiv:1705.08500.

[16] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On Calibration of Modern Neural Networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 70. PMLR, 2017, pp. 1321–1330.

[17] J. S. Boyle, A. Kascenas, P. Lok, M. Liakata, and A. Q. O'Neil, "Automated clinical coding using off-the-shelf large language models," in *Deep Generative Models for Health Workshop (DGM4H), NeurIPS 2023*, 2023. [Online]. Available: https://arxiv.org/abs/2310.06552

[18] K. Kwan, "Large language models are good medical coders, if provided with tools," arXiv preprint arXiv:2407.12849, 2024.

[19] Z. Yuan, H.-C. Shing, M. Strong, and C. Shivade, "Toward Reliable Clinical Coding with Language Models: Verification and Lightweight Adaptation," arXiv preprint arXiv:2510.07629, 2025.

[20] K. Krishna, A. Pavel, B. Schloss, J. P. Bigham, and Z. C. Lipton, "Extracting Structured Data from Physician-Patient Conversations by Predicting Noteworthy Utterances," in *Explainable AI in Healthcare and Medicine*, ser. Studies in Computational Intelligence. Springer International Publishing, 2021, pp. 155–169, arXiv:2007.07151.

[21] X. Ge, S. Murtaza, A. Cortez, and H. Alemzadeh, "EMSDialog: Synthetic multi-person emergency medical service dialogue generation from electronic patient care reports via multi-LLM agents," in *Findings of the Association for Computational Linguistics: ACL 2026*. San Diego, California, United States: Association for Computational Linguistics, Jul. 2026, pp. 35 081–35 110. [Online]. Available: https://aclanthology.org/2026.findings-acl.1751/