Predicting the Next Category and Region of a Visit: A Check-in-Level Multi-Task Study on Mobility Data

Vitor Hugo de Oliveira Silva
NESPeD-LAB
Universidade Federal de Viçosa
Florestal, MG, Brazil
vitor.h.oliveira@ufv.br

Germano Barcelos dos Santos
NESPeD-LAB
Universidade Federal de Viçosa
Florestal, MG, Brazil
germano.santos@ufv.br

Fabrício Aguiar Silva
NESPeD-LAB
Universidade Federal de Viçosa
Florestal, MG, Brazil
fabricio.asilva@ufv.br

***Abstract*—Location-based social networks record where people go and what they do, one check-in at a time. If we can anticipate the next visit, mobile and urban services can cache content or reserve capacity ahead of demand. Two coarse questions are usually enough: the category of the next visit and its region.**

We give each check-in its own vector instead of giving every place one fixed vector. With the same model, folds, windows, and training configuration, and only the input changed, this improves next-category prediction at every one of six datasets (five U.S. states and one non-U.S. city) and in every fold, by 0.23 to 6.29 points of macro-averaged F1, under one point at the three largest, and a paired test separates the two representations at five of the six.

We then ask whether one model can answer both questions in a single forward pass. On next region it outperforms a dedicated single-task model at the two datasets with the largest region vocabularies, by about one point, and at the other four it stays within a two-point margin registered before any result was read. On next category it outperforms the dedicated model at one dataset, and the other five differences are equivalent to zero within half a point. One model therefore serves both tasks at a cost bounded on both axes.

***Index Terms*—mobility data; next-category prediction; next-region prediction; multi-task learning; check-in-level representation; location-based social networks**

I. Introduction

Location-based social networks, such as Gowalla [1], let people share the places that they visit as check-ins, which together record how people move through a city. Individual mobility is highly predictable in principle [2], so a service that anticipates the next move can prepare ahead of time, caching content where a user is heading [3] or provisioning capacity at the destination before demand arrives. Handover predictions already let cellular services adapt in advance at the network level [4]; at the city scale, check-in mobility analysis is likewise a design input for mobility-aware services [5].

Predicting the next place exactly, a single point of interest (POI), is hard and often more than a service needs. Two coarser questions are usually enough: the next category, the type of place such as food or shopping, and the next region, the part of the city that a user is heading to. We study whether one model should learn both at once.

Sharing one representation across tasks has a cost: in multi-task learning (MTL), the shared parameters can converge to a compromise optimal for neither task, helping one while hurting the other [6]. Our earlier work reported no consistent multi-task advantage for the paired category tasks and attributed it, in part, to this effect [7]; the useful question is where sharing helps and what it costs.

We propose two enhancements to that earlier model. First, we build a check-in-level representation: instead of one fixed vector per place, each check-in gets its own vector that holds its context (the time, nearby places, and recent visits). This adds a fourth level, the check-in, beneath the place, region, and city levels of hierarchical graph infomax [8], [9] (Fig. 1). Second, we train one model that predicts the next category and the next region in a single forward pass, with a shared trunk (a cross-attention stack where the two tasks exchange semantic context) and a private spatial path for the region task (Fig. 2). We evaluate on two settings chosen to differ: five U.S. states of different sizes (Gowalla) and one non-U.S. city (Istanbul).[1]

Our contributions are as follows.

- **A check-in-level representation.** Under a controlled comparison in which only the input changes, at one random initialization, it improves next-category prediction at every one of the six datasets and in every one of their folds, by $+0.23$ to $+6.29$ points of macro-F1, under one point at the three largest datasets, and a paired test separates the two representations at five of the six (Table III). The complete system is also above every external baseline of Table II, on both tasks and at every dataset.
- **A single model for both tasks.** One model predicts the next category and the next region in one forward pass, reading both answers from a single saved model, so there is one artifact to train, version, and deploy. To our knowledge, fine-grained region as an end target of

[1]Code (model, representation, baselines, statistical tests): https://github.com/VitorHugoOli/PoiMtlNet/tree/mobiwac. Both data sources are public: the category-annotated Gowalla dump (https://figshare.com/articles/dataset/gowalla_data/22126586) and Massive-STEPS [10].

979-8-3195-0167-7/26/$31.00 ©2026 IEEE

equal standing is underexplored (Section II-B). The joint model outperforms a dedicated single-task region model at the two datasets with the largest region vocabularies and stays within a registered two-point margin at the other four (statistical non-inferiority, assessed with two one-sided tests). The joint model is also the larger model (Section VI). On category it outperforms the dedicated model at one dataset, and the five remaining differences are equivalent to zero within half a point, even though the dedicated category model receives the wider search (Table II). The pairing of the two region gains with the two largest region vocabularies is an observation and not a law (Section V-B).

II. Background and Related Work

A. From place embeddings to per-visit embeddings

A place embedding transforms a POI into a vector so that similar or nearby places sit close together in the vector space. Deep Graph Infomax (DGI) [9], a general self-supervised method for graphs, learns such vectors on a place network linked by similarity, time, and distance. Hierarchical Graph Infomax (HGI) [8] adds geographic structure, organizing the network as place, then region, then city. Both produce one vector per place, so two visits to the same coffee shop look identical to the model. Contextual check-in representations instead give each visit its own vector: CTLE [11], the closest example, learns a vector per visit by masking and reconstructing parts of a user's check-in sequence. CTLE is a sequence model, a Transformer that reads the check-in sequence itself; our representation, Check2HGI, remains a network model, keeping the hierarchical place-to-region-to-city network and the same infomax objective, now extended one level deeper, from individual places down to individual check-ins (Fig. 1). The order of a user's visits still enters the network, as links between consecutive check-ins (Section III-B).

We compare against CTLE and a feature-concatenation control to separate the combination from contextualization alone and from feature injection.

B. Predicting the next category and the next region

A large body of work predicts the next place that a user will visit, the exact POI, with recurrent and attention models such as DeepMove [12], STAN [13], and GETNext [14]. In contrast, we target two properties of the next visit, its category and its region, rather than the exact place. We choose them because they are what a mobility-aware service can act on, not to make the task easier; the region alone spans hundreds to several thousand classes. Predicting over a partition of the map is standard in the human-mobility literature, with a grid cell as the target [15]; we substitute official neighborhood-scale units. Our earlier work [7] paired static category classification with next-category prediction. This paper introduces the next-region task, adds the check-in-level representation, and reports, for each task, where the joint model outperforms a dedicated single-task model and where their difference stays within a stated bound (Section V-B).

The field increasingly models several granularities at once; in those systems, category and region are auxiliary signals that help a primary next-place task (MCMG [16], HMT-GRN [17]). We study the pair as the object itself. To our knowledge, fine-grained region as an end target of equal standing, rather than an auxiliary coarse grid cell, is underexplored. The nearest exceptions do not study our exact pairing: DRRGNN [18] forecasts a person's next activity region jointly with a mobility-intention label, but over regions discovered per person rather than a fixed citywide partition; a generative recommender [19] predicts category and region together, but only as auxiliary steps toward its next-place ranking.

C. Multi-task learning for mobility

MCARNN [20] jointly predicts activity and location, and a recent hierarchy-aware model continues the pattern [21]; in both, the location target is the exact place. The closest alternative to our setting is the cascade: predict the category first, then the place given the category [22]. CSLSL [23] is its current form, predicting in a chain (when, then what, then where), with location as the primary output and category as an intermediate step; CatDM [24] likewise uses the predicted category only to filter candidate places. We instead predict category and region in parallel, as end targets of equal standing in one model with a single forward pass, and we drop the next-place target entirely. On optimization, we are conservative by design: joint training with a fixed loss weighting is standard practice [6], and gradient-balancing methods (PCGrad [25], Nash-MTL [26]) rarely improve on a well-tuned fixed weighting with two tasks [27]. We confirmed this during development, on an earlier preparation of the data: a screen of nineteen loss and gradient balancers at their default configurations, at a single seed on Alabama and Florida, found none that improved on fixed equal weighting on both tasks at both datasets. The cosine similarity between the next-category and next-region updates on the shared trunk, measured on the reported joint model at four of the six datasets (Istanbul, Alabama, Arizona, and Florida), is equivalent to zero at every one of them, with per-dataset means within two thousandths of zero, so a balancer has no conflict to resolve. This is a finding for this pair of tasks, not a general rule.

III. Method

A. Problem and tasks

Given a user's time-ordered check-in history, we predict two properties of the next visit. The first is its *category*, one of the seven top-level categories that the Gowalla data distributes with its places: Community, Entertainment, Food, Nightlife, Outdoors, Shopping, and Travel. We map Istanbul's source labels onto the same seven. The second is its *region*, the place's census tract (for Istanbul, the *mahalle*, a municipal neighborhood). We frame next-region as classification over the dataset's regions, 520 to 8,501 of them (Table I).

The motivation is practical: the category says what type of place comes next, hence what a user will want; the region says where, hence where to prepare. The mobility literature

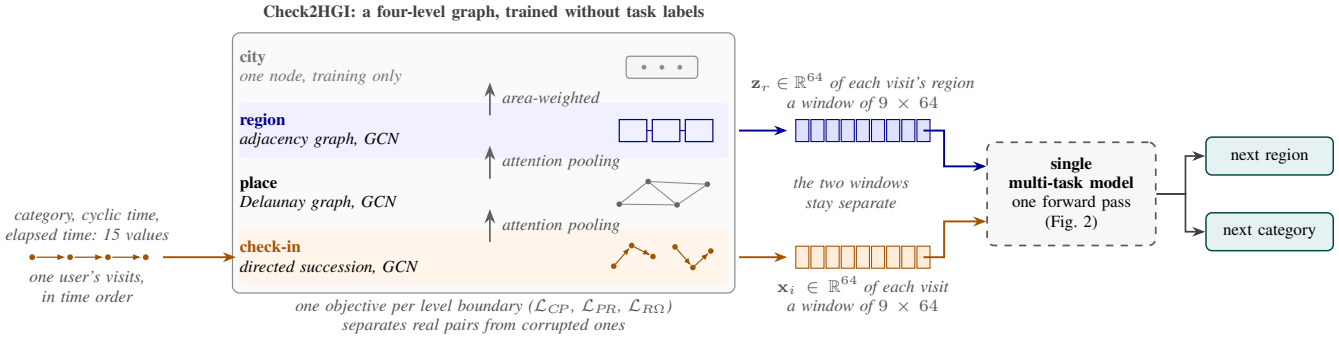


Fig. 1. From a check-in sequence to two predictions. Each visit is embedded in a four-level graph trained without task labels, and sliding nine-visit windows of the resulting per-visit and per-region vectors feed a single model.

supports such preparation: city services have long been built on location-based social network traces [28]. A recent analysis of tourist check-in mobility finds that visits concentrate on a few hub places, argues that crowds and demand therefore concentrate, and ties this structure to infrastructure planning and adaptive strategies [5]. That analysis reads past visits; we predict two properties of the next one, and aggregated over users, those predictions point to where demand is heading before it arrives. A census tract is a neighborhood, not a radio cell. We therefore scope our claims to neighborhood-level preparation: demand and load anticipation, caching content ahead of time, and capacity planning. Cell association and handover are radio-level decisions and remain out of scope. We build and evaluate no such service here, and Section VI states what these predictions would give such a service.

B. The check-in-level representation

We call this representation Check2HGI (Section II-A), and Fig. 1 shows the data flow.

We build a graph with four levels: the check-in, its place, the place's region, and the city. Edges connect each level to the one above it and link a user's consecutive check-ins with a weight that decays with the time gap. Nearby places are linked at the place level. Each visit's category, time of day, and day of week enter as input features of its node, together with four elapsed-time features: the log-scaled time since the user's previous visit and since their first visit, the same-day gap, and a first-visit indicator. Each is measured up to the visit itself, so a node describes the visit and the history preceding it, never anything that follows. The consecutive-visit edges run in one direction only, from an earlier visit to a later one, for the same reason: a target is predicted from a user's past, so the representation is built from the past alone. We train the graph mainly with an infomax objective [8], [9], plus two small label-free auxiliary terms (weights 0.3 and 0.1): a masked reconstruction of each place's aggregated category features, and an anchor to a place embedding pre-trained on the same data. The training never sees the next category or the next region. From the trained graph we extract one 64-dimensional vector per check-in.

C. The single multi-task model

One model reads a window of recent per-visit vectors and predicts the next visit's category and region at once (Fig. 2). Each task has its own input: the category task reads the window of per-visit vectors (the semantic stream), the region task the same window with each visit represented by the trained vector of its region node (the spatial stream). The region vectors are a fixed table, one vector per region, read from an earlier build of the same graph design. A per-region constant is the same in every window, so it cannot carry a window's target, and rebuilding the graph with directed edges moved the joint region result by 0.01 points at Alabama. Both pass through private per-task encoders into the shared trunk, a cross-attention stack of two blocks in which each stream reads the other's features while keeping its own feed-forward weights. The trunk feeds a category output and a region output, and the region output also keeps a private spatial path, a small branch inside the one model that bypasses the trunk.

The training loss is a fixed-weight sum of the two outputs, $L = 0.5\,L_{\text{cat}} + 0.5\,L_{\text{reg}}$, where L_{reg} is a plain unweighted cross-entropy loss and L_{cat} is a cross-entropy loss with logit adjustment [29]: $\tau \log P_{\text{train}}(y)$, with $\tau = 0.5$, is added to the category logits during training only, which shifts the decision boundary toward the balanced posterior that macro-F1 rewards. The weights and τ were selected once on validation during development and held fixed across all six datasets. The dedicated category model receives the same adjustment, so the comparison is not affected by it.

The joint model is larger than either dedicated model and than the two combined, and a forward pass costs more compute: about 4.2 million parameters at Alabama against 1.9 million for the two combined (5.2 against 2.8 at California). What the single model provides is operational rather than arithmetic (Section I).

IV. Experimental Setup

A. Data

Table I summarizes the six datasets. Five come from Gowalla, collected from 2009 to 2011, and cover the U.S. states of Alabama, Arizona, Florida, Texas, and California [1]. The sixth contains check-ins from Istanbul in the Massive-STEPS collection [10], which lets us examine the findings outside the United States.

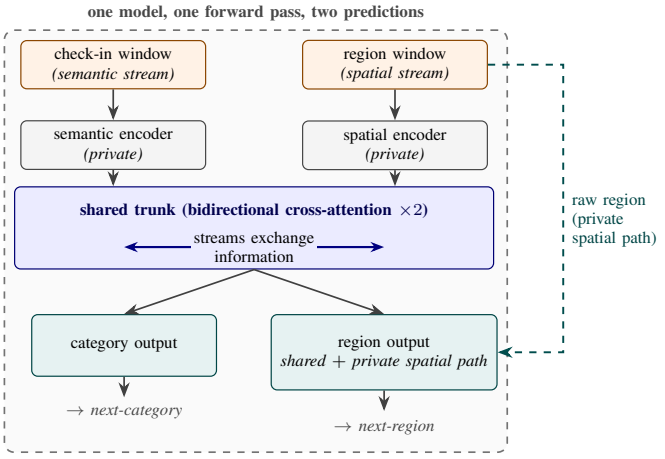


Fig. 2. The single multi-task model. A check-in (semantic) window and a region (spatial) window pass through private encoders into the shared trunk, the only place where the two tasks interact. The category output uses the trunk's output alone; the region output combines it with a private spatial path.

B. Windows, splitting, and the integrity of the representation

Windows. For each user with at least ten visits, we order the visits by time. We then form overlapping sliding windows of nine visits, one starting at each visit, and use the next visit as the target. Where several start positions share a target, we keep only the full-length window. The median time from the last visit in a window to its target ranges from 0.4 hours in Florida to 5.5 hours in Istanbul. Across the datasets, 5 to 27 percent of the targets occur more than 3 days later.

Splitting. We use stratified five-fold cross-validation and split the data by user. All windows from one user remain in the same fold. Therefore, the overlapping windows cannot cross the training and test sets, and the visits of a test user never appear in training. The split is by user and not by time, so the evaluation does not measure drift or the effect of sporadic events. The held-out fold provides the validation data, and we do not reserve a third split. We stratify the split by the next-category label because the seven classes are imbalanced (Table I).

Configuration search. The search was not uniform across the three model families. For the dedicated category model, the configuration was searched over five folds at Istanbul, Alabama and Arizona and on single folds at Texas. Florida and California carry the large-dataset value, which only the single-fold Texas screen tested, and the selected learning rate differs by dataset. For the joint model, batch size and the category-head learning rate were searched at Istanbul, Alabama, and Arizona over five folds and screened at Florida, with Texas and California carrying the configuration transferred from those searches. The dedicated region model uses one configuration throughout, with only the logit-adjustment strength tested and set to zero.

Whole-dataset training. The prediction models are trained without the validation users, but Check2HGI is trained once on the whole dataset and has seen their visits. Its training objective uses neither the next-category nor the next-region label. To check whether this gives the prediction models an advantage, we built a new representation for each fold using only its training users. This check covers three datasets at one seed, on an earlier build of the representation. The differences were -0.33 to $+0.01$ Acc@10 points for region and 0.00 to $+0.29$ macro-F1 points for category.

However, the region comparison uses region vectors directly. For category, a graph built only with training users has no visit vectors for validation users. We therefore used one vector per place and kept only windows whose input places occurred in training. These windows cover 67 to 87 percent of the validation data. Within this coverage, whole-dataset training did not meaningfully change the results.

C. Metrics and statistical tests

For the category task, we report macro-averaged F1 (macro-F1), the mean of the seven per-category F1 scores, so each category has equal weight. For the region task, we report accuracy at ten (Acc@10), the fraction of test visits whose true region is among the model's ten highest-scoring predictions. If the true region does not occur in the training data for that fold, the visit counts as an error. The reference point for this task is the dedicated model.

A non-significant difference does not provide evidence of a match. A written analysis plan, fixed during development and before any result was read, assigned a superiority test to next-category prediction and a non-inferiority test to next-region prediction. The plan did not define a superiority test for next-region prediction. Therefore, the two next-region gains in Section V-B are secondary results outside the plan. On next category the plan registered no equivalence margin, so a difference that fails the superiority test is reported as unresolved rather than as a match.

The plan registered a paired Wilcoxon signed-rank test on the matched fold differences. The final evaluation uses four seeds, each with its own user partition, so each configuration yields $4 \times 5 = 20$ fitted models, and the reported analysis is a paired t-test on the four per-seed means (n=4) with the 90% confidence interval of the paired difference. At this footing the exact one-sided Wilcoxon p-value cannot fall below 0.0625, which is why the t carries the verdict. The code release includes both tests and documents this departure from the plan.

A *seed* is one complete repetition of the five-fold experiment. It determines the random initialization and the user partition, so each seed produces a different division of the users. Within one seed, the compared models use the same partition. The paired t and TOST analyses compare the four per-seed means. We apply a Holm correction [30] across the six next-category comparisons and, separately, across the six next-region comparisons.

The non-inferiority claim states that the joint model is no worse than the dedicated model by more than a two-point margin. We test this claim with the two one-sided tests (TOST) procedure [31]. The analysis plan also fixed the two-point margin in advance. A mobility-aware service acts on which region will be busy, rather than on a single position in the

ranking (Section III-A). A two-point change in Acc@10 is below the level at which this service would behave differently. Across the four user partitions, the standard deviation of the paired difference ranges from 0.02 to 0.16 points.

D. Baselines

Every external system is run on our data, each with the adaptations its change of task requires: HMT-GRN, STAN, and POI-RGNN on our user-disjoint folds, and ReHDM [32] under its published chronological split.

Task comparisons. For next-category prediction we re-implement POI-RGNN [33]. It reads non-overlapping nine-visit windows, whereas our category models read overlapping ones. We also use a Markov model over the recent categories, with the best order selected per dataset, and, for region, a Markov-1 floor over region transitions. The primary next-region comparison is HMT-GRN [17], an end-to-end recurrent model. We keep its shared multi-task structure and add a region-transition prior built from the training data in each fold. During development, our own region output used such a prior, and a version built from the whole dataset inflated its region accuracy by 13 to 27 points, which is why the prior is built per fold. Only the HMT-GRN comparison model uses this prior, and the reported joint and dedicated models do not. We remove its graph components and hierarchical beam search, which support exact next-place prediction, so the resulting model predicts region as one of its original targets but is not a reproduction of the complete published system. STAN [13] is re-implemented with its own representations and sequences, its output adapted to rank candidate regions.

Representation controls. Two controls test whether the category gain comes from our representation design rather than from contextualization in general or from additional features: CTLE [11], the closest prior contextual check-in representation, and the standard place embedding (HGI [8]) concatenated with raw per-visit features (Section V-A). HGI is pre-trained once on the whole dataset, as our representation is, while CTLE is trained for each fold using training users only. All other choices, including the nine-visit window, were fixed during development, and the released code provides the full training configuration for every model (footnote 1).

V. Results

A. The check-in-level representation improves the category task

The check-in-level representation reaches a higher next-category macro-F1 than a standard place embedding (HGI [8]) at every dataset (Table III). Only the input representation differs, a vector per visit versus a vector per place. The check-in-level column is the dedicated category model of Table II at seed 0, and the place-level column swaps its input. The gaps run from $+0.23$ to $+6.29$ points and stay under one point at the three largest datasets, so what the comparison establishes is a consistent direction rather than a large effect.

The benefit is category-only by construction: the region task reads region-level vectors, not the per-visit vectors (Section III-C). CTLE [11], the closest prior contextual embedding, also gives each visit its own vector, but it pretrains on place identifiers and timestamps alone, so the category vocabulary never enters its training, whereas our graph reads each visit's category and time of day as input features. At Florida, we fine-tune CTLE together with the task model, using its authors' defaults and the same 64 dimensions and windowing as ours. It reaches 33.45 macro-F1 at its best epoch and 29.69 at its final one (an epoch is one pass over the training data), below the place embedding under the same best-epoch rule. With frozen weights, fed to the same single-task model, CTLE repeats the ordering at Alabama, Arizona, and Istanbul. These CTLE results establish the ordering between the two representation families rather than a difference against the values in Table III. A feature-concatenation control joins the place embedding with the same raw per-visit features that our graph reads (the category one-hot plus hour and day-of-week encodings) and feeds the result to the same single-task model. On the category axis, at Alabama, Arizona, and Florida this control raises the place embedding by $+2.0$, $+1.7$, and $+0.8$ macro-F1, against place-to-check-in gaps of $+1.62$, $+2.58$, and $+0.23$ at the same states (Table III). Most of the category difference is therefore already available in the raw per-visit features, and this control does not separate what the check-in-level representation adds beyond them on this axis.

B. One model, two tasks

One model serves both tasks, and every difference on both axes stays inside the bound its axis supports. On next region it outperforms the dedicated model at Texas and California, the two datasets with the most regions, and at the other four it stays within the two-point margin registered before any result was read (statistical non-inferiority, assessed with two one-sided tests). On next category it outperforms the dedicated model at Florida, and at every dataset the difference between the two is equivalent to zero within half a point, simultaneously across all six.

Half a point is worth placing against the two comparisons that give the category axis its scale. The input representation moves the same metric by $+0.23$ to $+6.29$ points, up to thirty-two times the largest difference the choice of architecture makes, and the joint model stands at least 3.06 points above the strongest external baseline at every dataset. Whether one model or two produce the category prediction is therefore the smallest of the three effects at work here. For scale: always predicting the most common category reaches between 5.7 and 7.3 macro-F1 depending on the dataset (the majority-class floor, lowest at Florida, whose category mix is the most even), and a random region top-ten guess is right at most about two percent of the time (Section IV-C).

Every reported model is one saved model per fold, read at its validation-selected epoch: each dedicated model at its task's best epoch, and the joint model at the epoch selected by the geometric mean of its two validation metrics, with both tasks

TABLE I

DATASET STATISTICS, ORDERED BY CHECK-IN COUNT. LENGTHS: PER-USER CHECK-IN COUNTS; DENSITY: 100 × CHECK-INS/(USERS × POIS); MAJORITY: SHARE OF NEXT-VISIT LABELS IN THE MOST COMMON CATEGORY. CHECK-INS, USERS, AND POIS COUNT THE FULL CORPUS; WINDOWS IS COMPUTED AFTER THE TEN-VISIT USER FILTER.

Dataset	Source	Check-ins	Users	POIs	Regions	Windows	Max len	Avg len	Density (%)	Majority (%)
AL	Gowalla	113,846	3,858	11,848	1,109	96,326	3,835	29.5	0.249	34.2
AZ	Gowalla	236,450	7,869	20,666	1,547	200,895	5,589	30.1	0.145	34.0
Istanbul	Massive-STEPS	462,615	23,694	29,816	520	271,666	817	19.5	0.065	33.4
FL	Gowalla	1,407,034	21,052	76,544	4,703	1,274,418	16,679	66.8	0.087	24.7
CA	Gowalla	3,171,380	37,090	169,145	8,501	2,925,466	14,855	85.5	0.051	32.7
TX	Gowalla	4,089,892	38,644	160,938	6,553	3,830,414	42,300	105.8	0.066	31.0

TABLE II

ONE MODEL, TWO TASKS: THE JOINT MODEL AGAINST THE DEDICATED SINGLE-TASK MODELS AND THE EXTERNAL BASELINES, ORDERED AS IN TABLE I (THE REGION COUNTS IN THE SECOND COLUMN DO NOT FOLLOW THAT ORDER). BOLD MARKS THE LARGEST VALUE IN EACH ROW AND BLOCK AND UNDERLINE THE SECOND, AND WHEN THE TWO LARGEST TIE AT THE PRINTED PRECISION BOTH ARE BOLD. BESIDE THE JOINT VALUE, ↑ MARKS AN IMPROVEMENT OVER THE DEDICATED MODEL THAT SURVIVES HOLM CORRECTION WITHIN ITS TASK FAMILY, AND, IN THE REGION BLOCK, ≈ MARKS A DIFFERENCE INSIDE THE TWO-POINT MARGIN REGISTERED BEFORE ANY RESULT WAS READ (TOST; SECTION V-B). CATEGORY CARRIES NO EQUIVALENCE MARK, BECAUSE THAT MARGIN WAS REGISTERED FOR THE REGION AXIS ONLY.

		Next-category (macro-F1)				Next-region (Acc@10)				
Dataset	Regions	Markov-K	POI-RGNN	Dedicated	Joint (ours)	HMT-GRN	ReHDM	STAN	Dedicated	Joint (ours)
AL	1,109	20.50	23.80	**30.77** ±0.07	30.59 ±0.07	57.05	65.38	60.72	**70.12** ±0.10	69.24 ±0.16$^{\approx}$
AZ	1,547	23.92	27.64	**34.57** ±0.04	**34.57** ±0.06	43.70	53.00	49.86	**59.48** ±0.07	59.04 ±0.22$^{\approx}$
Istanbul	520	24.55	30.12	35.34 ±0.01	**35.42** ±0.06	60.42	69.33	61.86	**75.16** ±0.01	75.08 ±0.05$^{\approx}$
FL	4,703	29.74	34.49	37.35 ±0.02	**37.55** ±0.07$^{\uparrow}$	63.74	64.49	72.99	**76.69** ±0.01	76.54 ±0.01$^{\approx}$
CA	8,501	27.58	31.78	**35.63** ±0.01	**35.63** ±0.02	49.61	50.26‡	58.52†	63.48 ±0.03	**64.54** ±0.04$^{\uparrow}$
TX	6,553	28.67	33.03	**36.33** ±0.01	36.19 ±0.04	53.85	48.81‡	61.67†	64.94 ±0.01	**66.15** ±0.07$^{\uparrow}$

HMT-GRN is scored on visits whose region appears in training (>99% of test visits in every dataset and fold). †STAN partial folds: TX 4/5, CA 2/5 (seed 0). ‡ReHDM at TX and CA: a single seed. Joint and dedicated entries: four seeds × five folds; ±: sd across seeds. Markov-K: strongest order per dataset. Protocols and search coverage: Sections IV-B and IV-D.

read from that one model, which is what a deployed system can serve. Reading each task at its own best epoch instead would describe no single saved model, and it is more favorable to the joint model, by at most 0.23 macro-F1 and 0.93 Acc@10 points at any one seed, enough to turn four further category comparisons and two further region comparisons into improvements that survive the same Holm correction. Every verdict in this paper is the stricter convention's.

On region (Acc@10), the joint model outperforms the dedicated model at Texas and California. The two datasets where it outperforms are the two with the largest region vocabularies, which is where the region task has the most classes to rank. The ordering is not monotone inside that pair, since California has more regions than Texas and a slightly smaller gain, and region count co-varies with corpus size across these states, so the grouping is read as an observation about where the benefit appears rather than as a law relating gain to region count. Why the benefit appears there is a separate question (Section VI).

On category, Florida is the one dataset where the difference survives the Holm correction across the six (+0.19; 90% CI +0.14 to +0.25; corrected $p = 0.011$), with 19 of the 20 folds favoring the joint model. The other five differences do not survive it, and because the equivalence margin was registered for the region axis only, they are reported by the bound their intervals support rather than as matches: Istanbul (+0.08; +0.01 to +0.15), Arizona (−0.00; −0.04 to +0.03), California (−0.00; −0.03 to +0.02), Texas (−0.13; −0.19 to −0.08) and Alabama (−0.19; −0.33 to −0.04). The intervals themselves carry what can be said about magnitude: the widest of them reaches 0.33 points from zero, at Alabama, and with a Bonferroni adjustment across the six comparisons the widest reaches 0.49, so all six differences stay within half a point of zero at once. The bound is read off the intervals rather than established by a further test.

On region, Texas (+1.21; +1.13 to +1.29; corrected $p = 0.00013$) and California (+1.06; +1.03 to +1.08; corrected $p < 10^{-4}$) survive the correction, with all 20 folds favoring the joint model at both. The remaining four stay within the registered two-point margin, and their direction is stated rather than rounded away: all four are deficits, at Alabama (−0.87; −1.00 to −0.75), Arizona (−0.44; −0.62 to −0.25), Florida (−0.16; −0.19 to −0.13) and Istanbul (−0.08; −0.16 to −0.002), and every one of the four intervals lies entirely below zero. A test in the reverse direction, applied post hoc to the same six region comparisons and corrected across them, resolves three of the four, and Istanbul's is not resolved. Each of the four stays inside the margin, but none of them is a tie.

The joint model is also above every external baseline reported, on both tasks, across all six datasets (Table II): on region, the primary region-native comparison HMT-GRN [17], STAN [13] trained from the raw check-ins, and a ReHDM reference [32]; on category, POI-RGNN [33] and the Markov-K floor. These externals run on their own embeddings, so this comparison also includes the representation advantage of Section V-A; the controlled comparison for the joint model remains the Dedicated column of Table II. The first-order Markov region floor of Section IV-D, computed under our windows and folds, reaches 51 to 72 Acc@10 across the datasets; the joint model exceeds it by 4.1 to 10.0 points on all six datasets.

That floor is also above the three external region systems at most datasets: HMT-GRN falls below it at all six, the ReHDM

TABLE III

NEXT-CATEGORY MACRO-F1: THE CHECK-IN-LEVEL REPRESENTATION AGAINST A STANDARD PLACE EMBEDDING (HGI), ORDERED BY CHECK-IN COUNT (TABLE I). SAME SINGLE-TASK MODEL, TRAINING CONFIGURATION, FOLDS, SLIDING WINDOWS, EPOCH BUDGET, AND LOGIT ADJUSTMENT IN BOTH MODELS, SO ONLY THE INPUT REPRESENTATION DIFFERS (MEAN AND FOLD SD OVER FIVE MATCHED FOLDS, SEED 0). BOLD MARKS THE HIGHER VALUE IN EACH ROW.

Dataset	Check-in level	Place level	Δ
AL	**30.77** $\pm$1.16	29.15 $\pm$0.84	$+1.62$
AZ	**34.51** $\pm$1.15	31.93 $\pm$1.08	$+2.58$
Istanbul	**35.35** $\pm$0.89	29.07 $\pm$0.63	$+6.29$
FL	**37.36** $\pm$0.42	37.13 $\pm$0.41	$+0.23$
CA	**35.62** $\pm$0.47	34.74 $\pm$0.41	$+0.88$
TX	**36.32** $\pm$0.51	35.33 $\pm$0.48	$+0.99$

All five folds favor the check-in-level representation at every dataset. The difference is significant under a paired t-test over the five folds at every dataset except FL ($p = 0.07$).

reference at three, and STAN at four. The floor reads a strong signal that our sliding windows expose, since they advance one visit at a time and the target region is the last visited region in 32.9 percent of the windows at Alabama, and the three systems do not meet it on equal terms (HMT-GRN on our data, folds, and initialization, STAN on our folds with its own embeddings and sequences, ReHDM under its own published protocol). We therefore treat the floor, not the external systems, as the reference the region task has to clear, and the dedicated and joint models are above it at every dataset.

At Istanbul, the non-U.S. city, both differences are within a tenth of a point of the dedicated models (category $+0.08$ macro-F1, region -0.08 Acc@10), so what repeats on a different continent and region unit is the bounded picture, with neither difference resolved as a gain.

VI. DISCUSSION AND LIMITATIONS

One model serves both tasks in one forward pass, above two dedicated models on next region at Texas and California and inside the registered two-point margin at the other four (Table II). The four region results inside that margin are all small deficits rather than ties, the largest of them 0.87 Acc@10 points at Alabama, so the trade is a measured one and not a free substitution.

The design gives the two tasks a shared representation while keeping a private spatial path for the region output, and the evidence here does not separate their contributions: where the trunk was isolated directly, the ablation moved both tasks by amounts that a screen of that size cannot distinguish from noise, and it was not run at the two datasets that carry the region advantage. What can be stated is a claim about the model rather than about transfer between its tasks.

A service could treat the model's top-ranked next regions as an anticipatory set: at California, ten regions out of 8,501 contain the true next region 64.54 percent of the time, and at Texas, ten out of 6,553 contain it 66.15 percent of the time (the joint model's Acc@10 of Table II, read as a shortlist hit rate). The geographic size of a miss is the quantity that would matter to such a service, and measuring it requires per-visit predictions that the evaluation path does not retain, so it is left to future work. This remains motivation, not a measured service result.

Four limits qualify these results. First, the representation is trained once over all places; a per-fold rebuild from training users only changed the results by at most 0.33 Acc@10 and 0.29 macro-F1 across three datasets at one seed, with the qualification on the category half recorded in Section IV-B. Second, epoch selection consults the fold that the score is then read on (Section IV-B), so every absolute score reported here is optimistic. The comparison between the joint model and the dedicated models is affected far less: the selection rule is the same for both models on the same folds, each model selected on its own validation objective (Section V-B), and for the category comparison the dedicated model receives the wider search (Section IV-B), which makes the reported category difference conservative where that search is the wider of the two. It does not follow that the bias cancels exactly. Third, each visit node in the representation graph draws on the visits that precede it, so a vector encodes the visit together with its own history; the graph does not pass information from a later visit back to an earlier one, in training or at readout, which is what keeps a node from carrying a feature of the target it is used to predict. Fourth, the joint model has more parameters than the two dedicated models together (Section III-C), so its region result at Texas and California may come from size rather than from sharing. The design does not separate the two.

Our representation is a fixed per-visit vector that any downstream model can consume. Moura et al. [5], whose structural analysis of tourist check-ins reads past visits only, name machine learning as a next step; this study moves in that direction.

VII. CONCLUSION

We tested whether one model can predict both the category and region of a user's next visit. The central concern was that sharing could help one task while hurting the other. The evidence does not attribute the outcome to sharing alone. On the category task the input representation moves the score more than the choice between one model and two: at every one of the six datasets and in every fold it improves next-category prediction, by $+0.23$ to $+6.29$ macro-F1 points. On the region task, the joint model is also the larger model, and the design does not separate size from sharing.

On that representation, one model reads both answers from a single saved model. On region it outperforms the dedicated model at Texas and California and stays within the two-point margin registered before any result was read at the other four. On category it outperforms the dedicated model at Florida, and the five remaining differences are equivalent to zero within half a point. The joint model also remains above every external baseline in Table II at all six datasets, by at least 3.55 Acc@10 points over the strongest region reference and 3.06 macro-F1 points over POI-RGNN on category.

These results do not mean that multi-task learning helps automatically. Whether the exchange of information between the

tasks adds anything beyond the representation is not separated by the evidence here (Section VI). When the representation preserves the context of each visit and the architecture keeps a private spatial path where the tasks differ, one model can predict both what a user will do next and where, in a single forward pass instead of two.

ACKNOWLEDGMENT

The authors would like to acknowledge the support of the Research Support Foundation of the State of Minas Gerais (FAPEMIG), CNPq and CAPES.

REFERENCES

[1] E. Cho, S. A. Myers, and J. Leskovec, "Friendship and mobility: User movement in location-based social networks," in *Proc. ACM SIGKDD (KDD)*. ACM, 2011, pp. 1082–1090.

[2] C. Song *et al.*, "Limits of predictability in human mobility," *Science*, vol. 327, no. 5968, pp. 1018–1021, 2010.

[3] E. Bastug, M. Bennis, and M. Debbah, "Living on the edge: The role of proactive caching in 5G wireless networks," *IEEE Commun. Mag.*, vol. 52, no. 8, pp. 82–89, 2014.

[4] C. L. Vielhaus *et al.*, "Handover predictions as an enabler for anticipatory service adaptations in next-generation cellular networks," in *Proc. ACM MobiWac*. ACM, 2022, pp. 19–27.

[5] D. L. L. Moura, A. L. L. Aquino, and A. A. F. Loureiro, "On the design of mobility-aware systems: A tourist's perspective," in *Proc. IEEE/ACM MSWiM*, 2025, pp. 667–674.

[6] R. Caruana, "Multitask learning," *Mach. Learn.*, vol. 28, no. 1, pp. 41–75, 1997.

[7] V. H. O. Silva *et al.*, "An investigation into multi-task learning for point-of-interest category classification and next-POI prediction," in *Anais do CBIC*, 2025.

[8] W. Huang *et al.*, "Learning urban region representations with POIs and hierarchical graph infomax," *ISPRS J. Photogramm. Remote Sens.*, vol. 196, pp. 134–145, 2023.

[9] P. Veličković *et al.*, "Deep graph infomax," in *Proc. ICLR*, 2019.

[10] W. Wongso, H. Xue, and F. D. Salim, "Massive-steps: Massive semantic trajectories for understanding POI check-ins — dataset and benchmarks," arXiv:2505.11239, 2025.

[11] Y. Lin *et al.*, "Pre-training context and time aware location embeddings from spatial-temporal trajectories for user next location prediction," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, 2021, pp. 4241–4248.

[12] J. Feng *et al.*, "DeepMove: Predicting human mobility with attentional recurrent networks," in *Proc. Web Conf. (WWW)*, 2018, pp. 1459–1468.

[13] Y. Luo, Q. Liu, and Z. Liu, "STAN: Spatio-temporal attention network for next location recommendation," in *Proc. Web Conf. (WWW)*, 2021, pp. 2177–2185.

[14] S. Yang, J. Liu, and K. Zhao, "GETNext: Trajectory flow map enhanced transformer for next POI recommendation," in *Proc. ACM SIGIR*, 2022, pp. 1144–1153.

[15] M. Luca *et al.*, "A survey on deep learning for human mobility," *ACM Comput. Surv.*, vol. 55, no. 1, pp. 7:1–7:44, 2021.

[16] Z. Sun *et al.*, "A multi-channel next POI recommendation framework with multi-granularity check-in signals," *ACM Trans. Inf. Syst.*, vol. 42, no. 1, pp. 1–28, 2024.

[17] N. Lim *et al.*, "Hierarchical multi-task graph recurrent network for next POI recommendation," in *Proc. ACM SIGIR*, 2022, pp. 1133–1143.

[18] N. Zhu *et al.*, "Predicting a person's next activity region with a dynamic region-relation-aware graph neural network," *ACM Trans. Knowl. Discov. Data*, vol. 16, no. 6, pp. 1–23, 2022.

[19] K. Sun and M. Xu, "Knowledge graph tokenization for behavior-aware generative next POI recommendation," arXiv:2509.12350, 2025.

[20] D. Liao *et al.*, "Predicting activity and location with multi-task context aware recurrent neural network," in *Proc. IJCAI*, 2018, pp. 3435–3441.

[21] Y. Wang *et al.*, "Hierarchy aware-based multi-task learning for user location prediction," *J. Supercomput.*, vol. 81, no. 11, 2025.

[22] J. Ye, Z. Zhu, and H. Cheng, "What's your next move: User activity prediction in location-based social networks," in *Proc. SIAM SDM*, 2013, pp. 171–179.

[23] Z. Huang *et al.*, "Human mobility prediction with causal and spatial-constrained multi-task network," *EPJ Data Sci.*, vol. 13, no. 1, p. 22, 2024.

[24] F. Yu *et al.*, "A category-aware deep model for successive POI recommendation on sparse check-in data," in *Proc. Web Conf. (WWW)*, 2020, pp. 1264–1274.

[25] T. Yu *et al.*, "Gradient surgery for multi-task learning," in *Proc. NeurIPS*, vol. 33, 2020.

[26] A. Navon *et al.*, "Multi-task learning as a bargaining game," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 162. PMLR, 17–23 Jul 2022, pp. 16 428–16 446.

[27] D. Xin *et al.*, "Do current multi-task optimization methods in deep learning even help?" in *Proc. NeurIPS*, vol. 35, 2022.

[28] T. H. Silva *et al.*, "Urban computing leveraging location-based social network data: A survey," *ACM Comput. Surv.*, vol. 52, no. 1, pp. 17:1–17:39, 2019.

[29] A. K. Menon *et al.*, "Long-tail learning via logit adjustment," in *Proc. ICLR*, 2021.

[30] S. Holm, "A simple sequentially rejective multiple test procedure," *Scand. J. Statist.*, vol. 6, no. 2, pp. 65–70, 1979.

[31] D. Lakens, "Equivalence tests: A practical primer for t tests, correlations, and meta-analyses," *Soc. Psychol. Personal. Sci.*, vol. 8, no. 4, pp. 355–362, 2017.

[32] X. Li *et al.*, "Beyond individual and point: Next POI recommendation via region-aware dynamic hypergraph with dual-level modeling," in *Proc. IJCAI*, 2025, pp. 3081–3089.

[33] C. G. S. Capanema *et al.*, "Combining recurrent and graph neural networks to predict the next place's category," *Ad Hoc Netw.*, vol. 138, p. 103016, 2023.