Learning Geolocations for Cold-Start and Hard-to-Resolve Addresses via Deep Metric Learning

Govind
Amazon
gvindmg@amazon.com

Saurabh Sohoney
Amazon
sohoneys@amazon.com

Abstract

With evergrowing digital adoption in the society and increasing demand for businesses to deliver to customers doorstep, the last mile hop of transportation planning poses unique challenges in emerging geographies with unstructured addresses. One of the crucial inputs to facilitate effective planning is the task of geolocating customer addresses. Existing systems operate by aggregating historical delivery locations or by resolving/matching addresses to known buildings and campuses to vend a high-precision geolocation. However, by design they fail to cater to a significant fraction of addresses which are new in the system and have inaccurate or missing building level information. We propose a framework to resolve these addresses (referred to as hard-to-resolve henceforth) to a shallower granularity termed as neighbourhood. Specifically, we propose a weakly supervised deep metric learning model to encode the geospatial semantics in address embeddings. We present empirical evaluation on India (IN) and the United Arab Emirates (UAE) hard-to-resolve addresses to show significant improvements in learning geolocations i.e., 22% (IN) & 55% (UAE) reduction in delivery defects (where learnt geocode is >Y meters[1] away from actual location), and 43% (IN) & 90% (UAE) reduction in 50th percentile (p50) distance between learnt and actual delivery locations over the existing production system.

1 Introduction and Motivation

Last Mile delivery planning systems aim to optimize the delivery experience for both customers and delivery associates when packages travel from the final delivery stations to customer doorsteps. One crucial input to this planning is the delivery location of customers. Customers provide information regarding their whereabouts through address text, the only mandatory input they need to provide while placing their order. The task of learning geolocation of addresses is commonly known as geocoding and it is challenging in emerging geographies because of two primary reasons – 1) Lack of standardisation in the address text in form of spelling variations, missing components and use of vernacular content, synonyms and abbreviations, 2) Large proportion of cold-start addresses to which too few or no deliveries have been made in the past. For instance, the address *Bank Colony Sheriguda, Ibrahimpatnam, Gandi Statue, 501510, Hyderabad, IN* does not contain any fine-grained details other than mentions of locality and landmark. Apart from the unstructured nature, addresses in emerging geographies tend to have inaccurate components such as *XX*[1] *Marina Bay One, Rawdat Al Reef, Abu Dhabi, UAE* contains wrong customer chosen district information (correct: *Al Reem Island*). It should be noted that the geocoding problem becomes trivial and simply reduces to aggregation of past delivery scans once there are successful deliveries to the address, irrespective of the address quality. The central theme of this work is to deal with customer addresses which have little or no delivery history along with missing or inaccurate components, also referred to as hard-to-resolve addresses here.

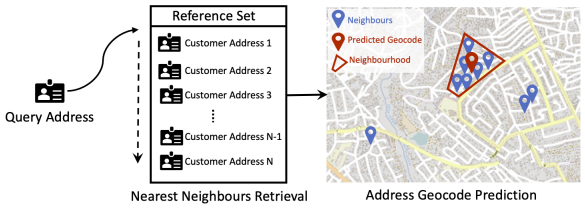


Figure 1: Address geocoding via nearest neighbours

In general, the address geocoding task is largely approached as entity matching or record linkage in natural language processing (NLP) where the idea is to match a query address to a reference set of addresses with known geocodes (Comber and Arribas-Bel, 2019; Lin et al., 2020; Li et al., 2022).

[1]In this paper, the exact values at few places are not revealed due to the business confidentiality reasons and finer address details are masked (X) to preserve customers' privacy.

These models target for a fine-grained (i.e. building, campus level) or exact matching of addresses and thus do not serve to a large fraction of addresses in emerging geographies which have missing or inaccurate building/campus information. Figure 1 illustrates a conceptual view of our geocoding pipeline. Rather than matching a query address to an individual address in reference set, we aim to retrieve its nearest neighbours. Using neighbourhood based approach we attempt to treat addresses not in isolation but in the view of their multiple neighbours, making it more robust to inconsistencies in hard-to-resolve addresses. The matched set of addresses can be used to vend a geocode and/or to jointly form a neighbourhood for the query address which can be vended as a approximate area to guide a delivery agent. In this paper, we focus on the experimental evaluation of geocoding task, and keep the neighbourhood polygons study as a work in the future.

We propose a novel deep metric learning based model to encode the geospatial distance semantics in address embeddings, in turn facilitating the retrieval of neighbours solely based on the address text. We pre-train the transformer based RoBERTa (Liu et al., 2019) model on address data and further employ a triplet network to learn quality address embeddings. A major proportion of hard-to-resolve addresses that this work targets, do not have any delivery history. Thus, our models draw supervision from past delivery scans while training only, and solely use address text (cold start) at the time of inference. In summary, our contributions are:

- We propose a deep metric learning based model to facilitate the encoding of geospatial distance semantics into address embeddings.
- We introduce a novel training data generation strategy to learn from geospatially rich addresses via weak supervision and transfer the knowledge to operate on cold-start addresses.
- To demonstrate the real-world impact of our model, we perform experiments on multiple emerging geographies (IN & the UAE).

This paper focuses on the downstream geocoding task, but learnt address embeddings can cater to other applications in the delivery planning space such as address correction, parsing, and learning neighbourhoods for package sorting.

2 Related Work

Short Text Geolocation Learning Geocoding short text (especially Tweets) has been an active area of research (Zheng et al., 2018). In (Hulden et al., 2015), authors propose a Naive Bayes classifier with kernel to learn the geospatial distribution of words and predict geolocation for tweets. (Paule et al., 2019) propose a weighted voting based nearest neighbours model to predict the location of traffic events. (Kulkarni et al., 2020) propose a neural network model with multi-level S2 (geospatial data structure) grids loss to learn tweets geolocation. Further (Qian et al., 2020) experiment with a seq2seq geocoding model to directly predict geohash string for Chinese addresses. (Li et al., 2019a) introduce GeoAttn model, which focuses on geolocation signals in the text and attends to the relevant Point-of-Interests (POIs) for location prediction. Although, most of these studies operate on a coarser level of geolocation (such as large geospatial grids, city) in contrast to the address geocoding task in e-commerce domain, which requires predicting within few meters of the customer doorstep to optimize delivery operations.

Entity Matching and Addresses In NLP, entity matching (or record linkage/deduplication) refers to the task of matching a query data instance to instances in a reference set (Hu et al., 2019). (Guo et al., 2016) propose a deep relevance matching model and more recently, the large language models for entity matching are explored by Ditto (Li et al., 2020) and dual objective fine-tuning of BERT (Peeters and Bizer, 2021).

Address geocoding has also been largely approached as entity matching task. (Comber and Arribas-Bel, 2019) propose to first parse the address text into address fields (unit, building, etc.), and then apply a pairwise matcher model to find a matching address in reference set and make a geocode prediction. (Lee et al., 2020) also implement a similar process where a rule-based parser and an SVM based matcher with building number interpolation are used for geocoding. Further, (Lin et al., 2020) and (Li et al., 2022) utilize deep learning based model for semantic matching of addresses. (Chen et al., 2021) propose a contrastive learning based address matcher for Chinese addresses while synthetically manipulating address texts to generate matching pairs. (Yang et al., 2019) propose to learn embedding for places and then uti-

lize them to train a supervised places deduplication model. In (Ganesan et al., 2021), authors propose a clustering based unsupervised model to learn POIs from the address data.

Deep Metric Learning Deep metric learning is being widely used on similarity retrieval tasks in both computer vision (CV) and NLP domains (Kaya and Bilge, 2019). (Hermans et al., 2017) apply triplet loss for person re-identification task and (Chen et al., 2020) introduce the contrastive learning of visual representations (SimCLR) for object detection. Sentence embeddings using siamese BERT networks are proposed to learn better downstream task specific embeddings (Reimers and Gurevych, 2019). SimCSE (Gao et al., 2021) and DeCLUTR (Gao et al., 2021) exploit contrastive learning to learn sentence representation in an unsupervised setting. In geospatial domain, Tile2Vec (Jean et al., 2019) and Hex2Vec (Woźniak and Szymański, 2021) explore embeddings learning of map tiles, whereas (Samano et al., 2020) explore the mobility data to learn regions representation.

To the best of our knowledge, none of the aforementioned studies target geocoding of hard-to-resolve addresses in emerging geographies. Further, a systematic way to impart geospatial distance semantics in address embeddings remains unexplored. Unstructured geographies pose a variety of challenges as discussed in the Section 1, making our contribution non-trivial and impactful.

3 Proposed Model

We adapt the K-Nearest Neighbours (K-NN) model (Altman, 1992) for the address geocoding task by using Kernel Density Estimation (KDE) (Parzen, 1962; Forman, 2021). Our workflow for geocode learning is illustrated in Figure 1. In essence, the K-NN model first retrieves the neighbourhood set $\mathbb{N}$ for an address a and then predicts its geocode by picking the geocode of the neighbour x with highest kernel density value. Equation 1 formulates the kernel density estimator P over the retrieved neighbours $\mathbb{N}$ where $K(x;h)$ is a Gaussian kernel with haversine metric. The bandwidth h works as a smoothing parameter, we chose h as 200 meters after manual validation over 25m to 400m.

$$P_h(x) = \frac{1}{|\mathbb{N}|h} \sum_{n \in \mathbb{N}} K(x - n; h) \qquad (1)$$

The absolute nearest neighbours search becomes very computationally expensive in higher dimensional input space. Thus, we employ approximate nearest neighbours search (Li et al., 2019b) and build an Annoy (Erik et al., 2018) index over the address embedding vectors to fetch neighbouring addresses from the reference set. One key difference here from the other address or entity matching systems (Lin et al., 2020; Li et al., 2022, 2020) is the flexibility as we are not restricting the match to a given building or campus, rather allowing a shallow matching on the full address text to arrive at a neighbourhood that can be of any size, shape and granularity. To this end, once the nearest neighbours are retrieved, we normalize their scores w.r.t. the maximum score and prune out the neighbours with low normalized score (below 0.25). This has an adaptive thresholding effect as all neighbours will be preserved if having more or less equal scores, and if there are disparity in scores then the low scored neighbours will get pruned. Also, we perform basic outlier removal of potentially incorrect neighbours via $mean \pm 2 * sd$ over latitude and longitude values to compute a neighbourhood polygon via convex hull. The geocode of the query address is computed using the described KDE model as a representative geocode of the neighbourhood. In this setting, quality representation of addresses are of utmost importance for retrieval of quality nearest neighbours. Thus, we propose a deep metric learning driven address representation learning approach in the following.

3.1 Deep Metric Learning

Deep distance metric learning (or simply, deep metric learning) aims to automatically construct task-specific distance metric from (weakly) supervised data by employing deep neural networks (Kaya and Bilge, 2019). The learned distance metric/pseudo-metric can then be used to perform various downstream tasks (e.g., information retrieval, clustering). In the context of addresses geocoding, the idealistic goal for the aforementioned neighbourhood retrieval problem is to fetch the true neighbours (i.e. to mimic geospatial distance semantics) for an address by using its text information only. Thus, we aim at learning an embedding transformation function $f_\theta(x) : R^I \rightarrow R^O$ which maps geospatially closer addresses from the input data manifold in R^I onto metrically close points in the output embedding space R^O (θ denotes parameter set). Similarly, f_θ should map geospatially far addresses in R^I onto metrically distant points in R^O.

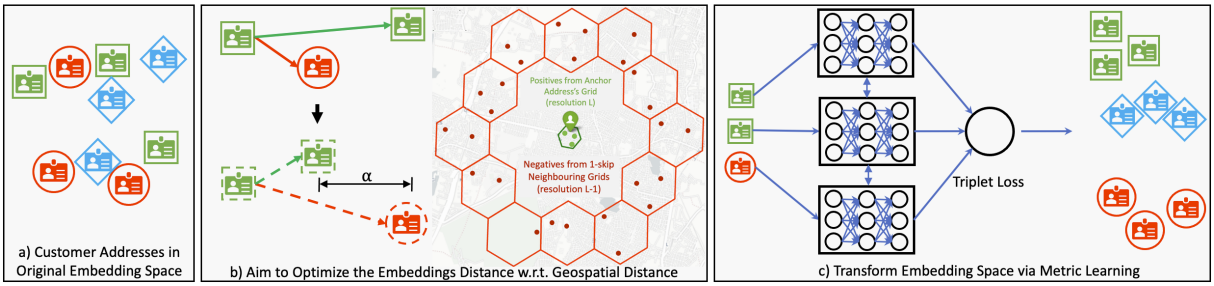


Figure 2: Deep metric learning on addresses to capture geospatial distance semantics

In the address domain, a key challenge with both the context-insensitive embeddings (e.g. FastText (Bojanowski et al., 2017)) or the contextualized embeddings (e.g. RoBERTa (Liu et al., 2019)) is the lack of understanding for geospatial distance semantics (cf. Section 4.2) as addresses do not follow a document or a paragraph like organization. Figure 2 depicts our adaptation of the deep metric learning workflow to learn quality address embeddings. As illustrated, we propose to systematically exploit geocodes of known addresses while training to give rise to geospatial distance semantics via weak supervision. To learn the transformation function f_θ, we choose RoBERTa as our base model as it has shown strong performance widely across multiple downstream NLP tasks (Liu et al., 2019).

Encoding Geospatial Semantics We employ contrastive learning approaches, specifically training via triplet loss. The triplet loss operates on triplets (x, x^+, x^-) of an anchor, a positive, and a negative instances. Equation 2 formulates the loss function with margin α and distance metric d. The objective here is to move the negative instance by distance margin α away from the anchor instance w.r.t. the positive instance. In our experiments, we chose margin 5 and Euclidean distance for triplet loss based on manual finetuning and practices in literature (Reimers and Gurevych, 2019).

$$L(x, x^+, x^-) = max(0, d(f_\theta(x), f_\theta(x^+)) - d(f_\theta(x), f_\theta(x^-)) + \alpha) \quad (2)$$

3.2 Training Data Generation

In classification, supervised metric learning algorithms use instance class labels (e.g. object, face identity) to generate the training data. However, manually labeling the matching/non-matching address pairs is very expensive and unscalable task. We employ historical delivery scans data to automatically generate the weakly labeled training pairs or the triplets. The address metric learning problem is now formulated as an optimization problem where we seek to find the parameters θ of function f_θ that optimize a objective function (i.e. triplet loss) measuring the agreement with training data.

Ideally, positive addresses for an address should be sampled from the absolute geospatial neighbours within some small β^+ distance and negatives should be sampled from the addresses which are relatively far β^- away. Here, the limitation is costly computation of haversine distance of each address to every other address, further even using some spatial data structure such as Ball Tree (Omohundro, 1989) involves significant computation overhead. To overcome this, we propose to use H3 geospatial indexing[2] system as an approximate solution to retrieve positive and negative addresses in a more intelligent manner. H3 is a hierarchical spatial data structure which subdivides the space into buckets of hexagonal grid shape. Every hexagonal grid has seven child grids below it in the hierarchy, thus, a hexagon of resolution L have 7 child hexagons of resolution $L + 1$ and so on (cf. Appendix C). For instance, $L = 10$ hexagon has edges of length 66m, and the children ($L = 11$) have 25m edges.

For an address, T positive addresses are sampled from its H3 grid of level L. T negative addresses are sampled from the ring of parent's (i.e. level $L - 1$) 1-skip neighbouring grids as shown in Figure 2b. To this end, we generate triplets by varying the resolution ($L \in [11, 10, 9]$) for positive samples (and consequently for negatives). The motivation behind including triplets with varying resolution is to compile a more diverse training data where triplets can encode very a fine-grained as well as a coarse grained comparison of addresses. As addresses in close vicinity tend to differ only in the header part, we generate another T triplets where negatives are sampled from the city level to enforce sufficient focus on the tail address components.

[2] H3 geospatial index https://github.com/uber/h3

4 Experimental Evaluation

We evaluate the learnt embeddings intrinsically and on the downstream geocoding task.

4.1 Experimental Setup

We experiment with IN and the UAE addresses. For each of the dataset we use large unlabelled address text to pre-train the RoBERTa model and use historical delivery scans data to generate weak supervision for metric learning. We operate on the last few years of data which are worth hundreds of millions of shipments and tens of millions of unique addresses. We do minimal preprocessing of the address text by replacing repeated space and punctuations to single character. For evaluation, a few weeks of out-of-time network wide shipments are considered where learnt geolocations are compared against the observed delivery scans (marked by delivery associates). As our solution is targeted towards hard-to-resolve cases (i.e., production baselines couldn't vend any confident geocode and fall back to postal code/locality centroids), we only run our pipeline for this particular subset. Note that due to confidentiality reasons, we cannot reveal the actual proportion of hard-to-resolve addresses however, they are considerably high for the emerging geographies such as IN & the UAE, which is why improvements on this subset result in large amount of savings network-wide.

Deep Metric Learning Data Set For metric learning experiments, we only consider the addresses with at least H historical scans[1] to be more confident on the actual location. We take a stratified sample w.r.t. grids to have better representation of addresses across a geography and to not skew the learning disproportionately towards high density metropolitan areas. We generate 2*T triplets[1] for an anchor address as explained in Section 3.2. To this end, we get a total of 37M triplets for IN and 7M triplets for the UAE.

Model Configurations and Baselines We perform extensive experiments on the task of geolocating hard-to-resolve addresses across various underlying models. We set up the current production geocoding system on the considered hard-to-resolve test set and report relative improvements over it. Due to the complex nature of these addresses, the baseline reduces to simply the centroid at postal code or locality level. For a better comparative analysis, we also consider a context-insensitive model based on FastText, which is a skip-gram model trained with character n-grams of size 3-5 and window size of 8 for 10 epochs on address data. Among the transformer models, we have two groups – 1) The first group includes RoBERTa-General which is the general purpose English RoBERTa-base model, and RoBERTa-Address (6 layers) is trained from scratch on address data; 2) The second group is based on metric learning framework. RoBERTa-Triplet is trained on triplets generated by sampling positives at single fixed H3 resolution ($L = 11$) only and negatives are sampled only from the city level. In contrast, RoBERTa-Triplet-H3 is trained using the proposed training data generation strategy, which operates at multiple H3 resolutions to generate better quality triplets (cf. Section 3.2). These two models are fine-tuned over RoBERTa-Address. The final address embedding vector is computed via mean pooling over token embeddings of the final layer.

Pre-training Address Language Model As addresses have quite different vocabulary and domain than general English text, we train from scratch the geography specific RoBERTa models (6 layers) with masked language modeling (MLM) objective on addresses data (tens of millions). We train Byte-Pair Encoding tokenizers with vocabulary size of 52K. The model training with sequence length of 100 and batch size of 64 for 10 epochs takes around 49 hours on 4 Tesla V100 GPUs.

4.2 Assessing Embeddings Quality

To intrinsically measure the geospatial distance semantics captured in address embeddings, we compute cosine similarity co-relation on address pairs. A test set of 0.5M pairs is compiled by sampling positive pairs (score 1) from the same H3 grid of resolution 9 and negatives (score 0) are sampled from city level (equal + & - pairs). Further, to evaluate more complex relationship among addresses, we generate a set of 0.5M triplets constrained by

	Pearson		Triplet Acc	
Model	**IN**	**UAE**	**IN**	**UAE**
FastText	0.56	0.66	84.23	86.91
RoBERTa-General	0.35	0.45	70.42	75.76
RoBERTa-Address	0.63	0.68	85.78	87.02
RoBERTa-Triplet	0.76	0.75	91.33	91.79
RoBERTa-Triplet-H3	**0.81**	**0.84**	**93.92**	**95.54**

Table 1: Address pairs cosine similarity co-relation (cf. Appendix A for density plots) and Triplet accuracy

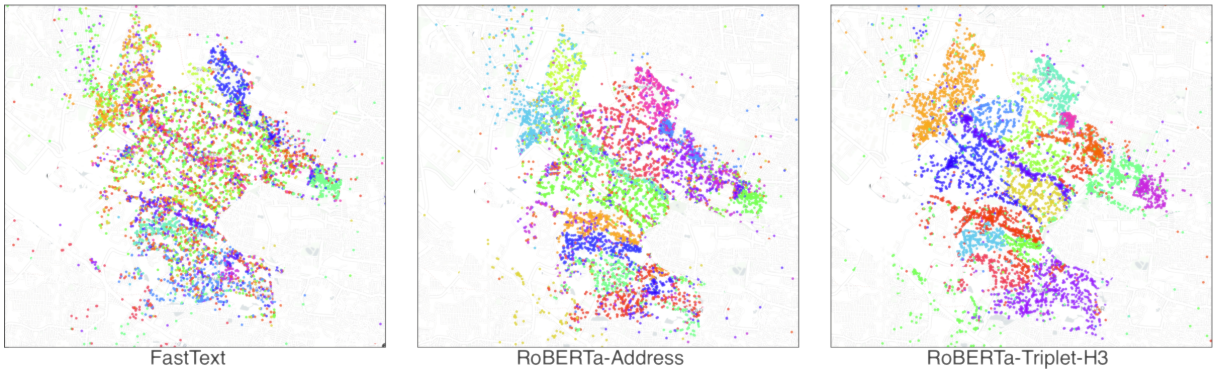


Figure 3: Clustering the addresses using different embeddings and visualizing via their geocodes (Note: background maps are modified and blurred to preserve customers' privacy)

	>Y DPMO (↓ %)		p25 (↓ %)		p50(↓ %)		p95(↓ %)	
Geocoding Model	IN	UAE	IN	UAE	IN	UAE	IN	UAE
FastText	19.3%	48.9%	84.0%	94.3%	35.2%	82.4%	**34.6%**	**61.6%**
RoBERTa-General	9.8%	34.3%	52.1%	88.9%	9.1%	60.1%	-44.2%	31.8%
RoBERTa-Address	20.9%	47.4%	86.2%	94.0%	**43.6%**	80.2%	33.2%	60.5%
RoBERTa-Triplet	21.0%	52.1%	85.7%	95.5%	41.1%	86.7%	31.5%	60.1%
RoBERTa-Triplet-H3	**22.0%**	**54.6%**	**88.6%**	**96.4%**	42.8%	**90.3%**	32.2%	53.1%

Table 2: Geocoding metrics relative to the production baseline on shipments against hard-to-resolve addresses

the only condition that anchor will be geospatially closer to the positive than the negative. Then the triplet accuracy is computed to evaluate if embeddings pass the same criterion using cosine distance. Table 1 reports Pearson co-relation and the triplet accuracy metrics and we observe that the metric learning based models outperform others by a large margin (cf. Appendix A for density plots).

We also do a qualitative analysis by clustering (K-means with K=20) the addresses using their embeddings and visualizing them via their geocodes (cf. Figure 3 for 50K addresses in an IN postal code). The motivation is that embeddings which capture quality geospatial distance semantics will result in smoother clusters by facilitating the grouping of geospatially closer addresses. We observe that FastText based embeddings produce clusters with very high overlaps. In contrast, RoBERTa-Triplet-H3 embeddings facilitate smoother boundary clusters because of better geospatial distance semantic understanding. RoBERTa-Triplet-H3 embeddings clusters' quality can also be seen slightly improving over the RoBERTa-Address. This is also visible in Silhouette scores which are 0.02, 0.08, and 0.13 for FastText, RoBERTa-Address, and RoBERTa-Triplet-H3 respectively. The observed geospatial distance semantics are beneficial for multiple downstream tasks such as address correction, package sortation, and address geocoding.

4.3 Geolocating Hard-to-Resolve Addresses

We compute neighbourhood level geocodes via KDE over the retrieved neighbours as illustrated in Figure 1 and serve to guide the drivers to a closer proximity in the absence of any better geocode. Table 2 presents experimental results via various geocoding metrics relative to the production baseline on shipments for the chosen test period. DPMO (Defects Per Million Opportunities) measures the number of prediction falling beyond Y[1] meters normalized to a million. The percentile metrics (p25, p50, p95) capture the distribution of error distances (actual vs predicted geocode) on the test set. A superior model shall lead to higher reductions in these metrics.

It can be observed from Table 2 that the proposed model based on deep metric learning outperforms the production baseline by a substantial margin as well as stands superior in comparison to other baselines i.e. FastText and basic Transformer models. The poor performance of RoBERTa-General model is due to its training on general purpose English text only. It can also be seen that RoBERTa-Triplet-H3 improves over RoBERTa-Triplet by a large margin, which can be directly attributed to the importance of our proposed training data generation strategy. Overall, we observe an improvement of 22% in DPMO for IN and 54% for the UAE (cf. Appendix B for geocoding anecdotes). This reduction in num-

ber of defects is directly translatable to the saved operational cost arising from delivery defects. It should be noted that addresses in IN & the UAE are quite different in nature, thus, improved metrics confirm the wide applicability of our framework. Further, IN has much bigger scale and more diverse addresses than the UAE, which manifests in our results with larger improvements in the UAE. We performed a ablation study by experimenting without adaptive thesholding (cf. Section 3) and observed degraded performance across models (IN DPMO became 13% for FastText and 16% for RoBERTa-Triplet-H3). It is also worth pointing out that the proposed model is trained with weak supervision and does not have a dependency on any manually curated ground truth or the address parsing models.

5 Conclusion and Future Work

In this work, we presented an efficient nearest neighbours & deep metric learning based approach to perform the address geocoding and facilitate the capturing of geospatial distance semantics in address embeddings. We intrinsically observe quantifiable improvements in address embeddings quality. Encouraging results from offline experiments suggest an immediate improvement in serving hard-to-resolve addresses. Our model operates solely using address text at the inference time, and is trained without any manually curated labels making it scalable across emerging geographies such as IN, the UAE, Mexico, and Saudi Arabia.

We plan to perform online experiments and extend our models to multi-lingual addresses in order to deal with prevalent issues like code switching in emerging geographies. We also would like to enhance our negative mining strategies and explore a pairwise cross encoder model to filter out the poorly retrieved neighbours. Retrieval of addresses from the neighbourhood can power many other downstream applications such as address component correction, address auto-complete suggestions, and optimizing delivery station assignment. We plan to explore geospatial constraints aware neighbourhood learning (e.g., to ensure neighbourhoods do not cross natural obstacles such as water bodies and highways).

References

Naomi S Altman. 1992. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3):175–185.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Jian Chen, Jianpeng Chen, Xiangrong She, Jian Mao, and Gang Chen. 2021. Deep contrast learning approach for address semantic matching. *Applied Sciences*, 11(16):7608.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. *CoRR*, abs/2002.05709.

Sam Comber and Daniel Arribas-Bel. 2019. Machine learning innovations in address matching: A practical comparison of word2vec and crfs. *Transactions in GIS*, 23(2):334–348.

Bernhardsson Erik et al. 2018. Annoy (approximate nearest neighbors oh yeah): Approximate nearest neighbors in c++/python optimized for memory usage and loading/saving to disk. *https://github.com/spotify/annoy*.

George Forman. 2021. Getting your package to the right place: Supervised machine learning for geolocation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 403–419. Springer.

Abhinav Ganesan, Anubhav Gupta, and Jose Mathew. 2021. Mining points of interest via address embeddings: An unsupervised approach. In *Proceedings of the 5th ACM SIGSPATIAL International Workshop on Location-Based Recommendations, Geosocial Networks and Geoadvertising*, LocalRec '21, New York, NY, USA. Association for Computing Machinery.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM international on conference on information and knowledge management*, pages 55–64.

Alexander Hermans, Lucas Beyer, and Bastian Leibe. 2017. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*.

Weiwei Hu, Anhong Dang, and Ying Tan. 2019. A survey of state-of-the-art short text matching algorithms. In *Data Mining and Big Data*, pages 211–219, Singapore. Springer Singapore.

Mans Hulden, Miikka Silfverberg, and Jerid Francom. 2015. Kernel density estimation for text-based geolocation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Neal Jean, Sherrie Wang, Anshul Samar, George Azzari, David Lobell, and Stefano Ermon. 2019. Tile2vec: Unsupervised representation learning for spatially distributed data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3967–3974.

Mahmut Kaya and Hasan Şakir Bilge. 2019. Deep metric learning: A survey. *Symmetry*, 11(9):1066.

Sayali Kulkarni, Shailee Jain, Mohammad Javad Hosseini, Jason Baldridge, Eugene Ie, and Li Zhang. 2020. Spatial language representation with multi-level geocoding. *arXiv preprint arXiv:2008.09236*.

Kangjae Lee, Alexis Richard C Claridades, and Jiyeong Lee. 2020. Improving a street-based geocoding algorithm using machine learning techniques. *Applied Sciences*, 10(16):5628.

Fangfang Li, Yiheng Lu, Xingliang Mao, Junwen Duan, and Xiyao Liu. 2022. Multi-task deep learning model based on hierarchical relations of address elements for semantic address matching. *Neural Comput. Appl.*, 34(11):8919–8931.

Sha Li, Chao Zhang, Dongming Lei, Ji Li, and Jiawei Han. 2019a. *GeoAttn: Localization of Social Media Messages via Attentional Memory Network*, pages 64–72. Proceedings of the 2019 SIAM International Conference on Data Mining (SDM).

Wen Li, Ying Zhang, Yifang Sun, Wei Wang, Mingjie Li, Wenjie Zhang, and Xuemin Lin. 2019b. Approximate nearest neighbor search on high dimensional data—experiments, analyses, and improvement. *IEEE Transactions on Knowledge and Data Engineering*, 32(8):1475–1488.

Yuliang Li, Jinfeng Li, Yoshihiko Suhara, AnHai Doan, and Wang-Chiew Tan. 2020. Deep entity matching with pre-trained language models. *Proc. VLDB Endow.*, 14(1):50–60.

Yue Lin, Mengjun Kang, Yuyang Wu, Qingyun Du, and Tao Liu. 2020. A deep learning architecture for semantic address matching. *International Journal of Geographical Information Science*, 34(3):559–576.

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

Stephen M. Omohundro. 1989. Five balltree construction algorithms. Technical report.

Emanuel Parzen. 1962. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076.

Jorge David Gonzalez Paule, Yeran Sun, and Yashar Moshfeghi. 2019. On fine-grained geolocalisation of tweets and real-time traffic incident detection. *Information Processing & Management*, 56(3):1119–1132.

Ralph Peeters and Christian Bizer. 2021. Dual-objective fine-tuning of bert for entity matching. *Proceedings of the VLDB Endowment*, 14(10):1913–1921.

Chunyao Qian, Chao Yi, Chengqi Cheng, Guoliang Pu, and Jiashu Liu. 2020. A coarse-to-fine model for geolocating chinese addresses. *ISPRS International Journal of Geo-Information*, 9(12):698.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Noe Samano, Mengjie Zhou, and Andrew Calway. 2020. You are here: Geolocation by embedding maps and images. In *Computer Vision – ECCV 2020*, pages 502–518, Cham. Springer International Publishing.

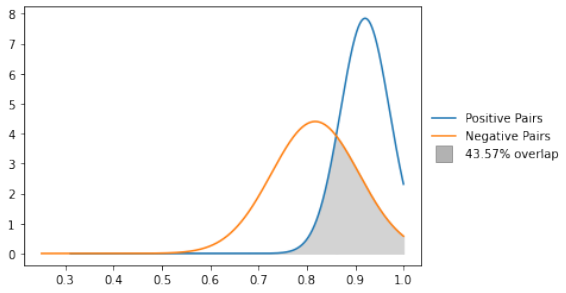
Szymon Woźniak and Piotr Szymański. 2021. *Hex2vec: Context-Aware Embedding H3 Hexagons with OpenStreetMap Tags*, page 61–71. Association for Computing Machinery, New York, NY, USA.

Carl Yang, Do Huy Hoang, Tomas Mikolov, and Jiawei Han. 2019. Place deduplication with embeddings. In *The World Wide Web Conference*, pages 3420–3426.

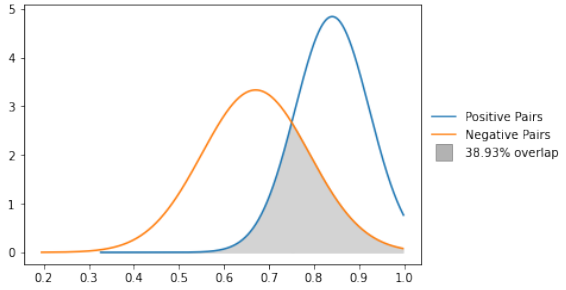
Xin Zheng, Jialong Han, and Aixin Sun. 2018. A survey of location prediction on twitter. *IEEE Transactions on Knowledge and Data Engineering*, 30(9):1652–1671.

A Cosine Similarity Density Plots

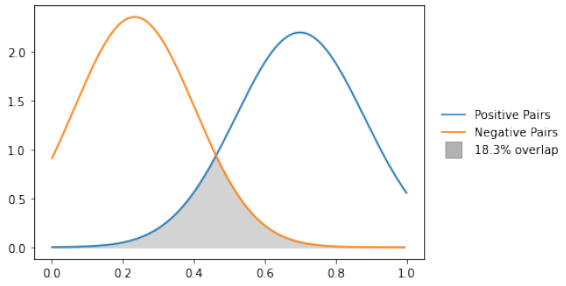
Figure 4 depicts the cosine similarity density plots for positive (label 1) and negative (label 0) address pairs in the test set (cf. Section 4.2). The x-axis represents cosine similarity values and it can be seen in Figure 4 that RoBERTa-Triplet-H3 model segregates well positives from negatives with least overlap between the two density curves (cf. Fig. 4c) in comparison to others (cf. Fig. 4a,b).



(a) FastText



(b) RoBERTa-Address

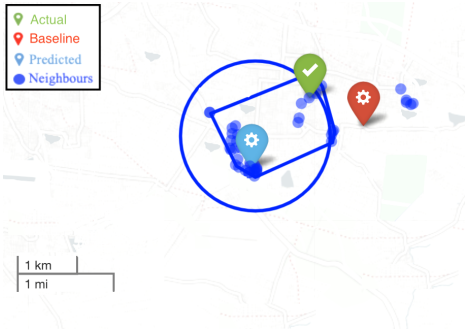


(c) RoBERTa-Triplet-H3

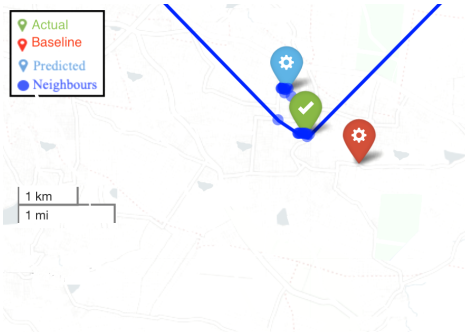
Figure 4: Density curves of positive (label=1) and negative (label=0) address pairs

B Anecdotes on Geocoding

Figure 5 depicts geocodes of the retrieved neighbours (along with neighbourhood polygons/circles) and the predicted geocode by various models for a hard-to-resolve address *X-X-X , Shivalayam Nagar , 500070 , Hyderabad , Telangana*. The masked information (*X-X-X*) here is the house number, which carry some relevance for geocoding but usually noisy and do not follow a standard pattern. It is a hard address as it is relatively sparse with no landmark information and the locality name is misspelled (*Shivalayam* instead of *Sachivalayam*). *Shivalayam* means Temple whereas *Sachivalayam* means Government Admin Office. There exist no *Shivalayam Nagar* in *500070 , Hyderabad*. We observe that the FastText model struggles to retrieve good quality neighbour addresses. RoBERTa-Address model utilizes the context and retrieves few good addresses but at the same time many poor matches too. The RoBERTa-Triplet-H3 model utilizes the contextual information best along with house number in address header to be resilient towards wrong locality name. It produces a quality set of neighbouring addresses to bring the predicted geocode as close as 39m to the actual location.



(a) FastText (prediction within 1482m)



(b) RoBERTa-Address (prediction within 792m)

(c) RoBERTa-Triplet-H3 (prediction within 39m)

Figure 5: Retrieved nearest neighbour addresses by various models for an example address: *X-X-X Shivalayam Nagar , 500070 , Hyderabad , Telangana*. The current production baseline vends a geocode 986m away and we vend within 39m. (Note: background maps are modified and blurred to preserve customers' privacy)

C H3 Hexagonal Grids

Table 3 reports size of hexagon grids with respect to various H3 index resolution levels, and Figure 6 illustrates the hierarchical relation between grids. The geographical containment of children by a parent is approximate while the logical containment in the index is exact. We choose H3 over other geospatial indices such as Geohash because of the benefits observed via the symmetry of hexagonal shape in contrast to squares/triangles which have neighbors at varying distances in different directions.

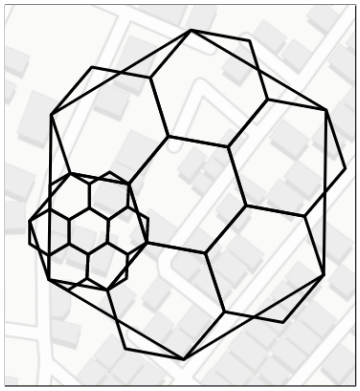


Figure 6: H3 parent and child hexagonal grids hierarchy

H3 Resolution	Edge (meters)	Diagonal (meters)
0	11,07,712.6	22,15,425.2
1	4,18,676.0	8,37,352.0
2	1,58,244.7	3,16,489.3
3	59,810.9	1,19,621.7
4	22,606.4	45,212.8
5	8,544.4	17,088.8
6	3,229.5	6,459.0
7	1,220.6	2,441.3
8	461.4	922.7
9	174.4	348.8
10	65.9	131.8
11	24.9	49.8
12	9.4	18.8
13	3.6	7.1
14	1.3	2.7
15	0.5	1.0

Table 3: H3 hexagonal grid edge and diameter sizes w.r.t. the resolution levels