# Introducing OpenStreetMap user embeddings: Promising steps toward automated vandalism and community detection

Yinxiao Li[1,*] and Jennings Anderson[2]

[1] Facebook, Boston, USA; yinxiaoli@fb.com
[2] YetiGeoLabs, Montana, USA; jennings.anderson@gmail.com

* Author to whom correspondence should be addressed.
This abstract was accepted to the Academic Track of the State of the Map 2021 Conference after peer-review.

With more than 11B edits from 1.6M unique mappers and openly editable by anyone, the OpenStreetMap (OSM) database inevitably contains vandalism. Our approach to detecting it leverages the analytical power and scalability of machine learning through OSM user embeddings. Embeddings are effective in capturing semantic entity similarities that are not explicitly represented by the data. Since word embeddings were first introduced based on the assumption that words adjacent to each other share similar meanings [1, 2], the concept of embeddings has been extended beyond word representations to any entity, so long as one can produce a meaningful sequence of the entities. Therefore, we build OSM user embeddings with mappers as entities by constructing sequences of mappers based on shared editing histories and similar behaviors.

Development of automated vandalism detection methods in OSM has been slow in part because there is no published corpus of bad or vandalized edits from which to train and validate [3]. Vandalized name attributes are especially problematic because this text is rendered on the basemap. The most infamous instance of this type of vandalism was the changing of "New York City" to an ethnic slur; this name attribute was subsequently rendered on maps drawing from OSM data [4]. As part of this work, we construct and make available the first OSM vandalism corpus for the name attribute of OSM features. Potential examples of vandalism are collected from the OSM Changeset Analyzer (OSMCha) web-based validation tool. These records are then manually reviewed by the Facebook mapping team to identify egregious name changes. Negative samples (non-vandalism) were randomly sampled from a previously validated vandalism-free snapshot of OSM. All of our examples are extracted from OSM data only, no external or additional conflated data sources.

To construct meaningful sequences of OSM users where adjacent users share similar mapping patterns, we analyzed the edit history of every OSM object and the temporal/semantic editing patterns of individual mappers. These sequences were then fed into a word2vec skip-gram model to train OSM user embeddings.

Li, Y. & Anderson, J. (2021). Introducing OpenStreetMap user embeddings: Promising steps toward automated vandalism and community detection
In: Minghini, M., Ludwig, C., Anderson, J., Mooney, P., Grinberger, A.Y. (Eds.). Proceedings of the Academic Track at the State of the Map 2021 Online Conference, July 09-11 2021, 23-26. Available at https://zenodo.org/communities/sotm-2021
DOI: 10.5281/zenodo.5112241

Shared object editing histories are sequences of OSM users who have edited the same object, in chronological order of editing. These sequences represent mappers who share interest in the same objects on the map. This yields 2B sequences of mappers.

Semantic and temporal mapping patterns are sequences of OSM users that have shared editing characteristics with regard to how and when they edit the map. Starting with *changesets*, we extract the following keys for each OSM element edited in a given changeset: `addr:country`, `admin_level`, `amenity`, `building`, `highway`, `natural`, `place`, `source`. Each of these attributes are not always present, but were chosen based on empirically observed high-level editing patterns among users. Additionally, we extract the following metadata: the presence of `name` tag, the `version` number, the editing software (e.g. iD editor, JOSM), and any hashtags (possibly denoting specific mapping campaigns). Finally, we group all of these edits by two types of temporal patterns: first, the date of the changeset, and second, the hour of the week of the changeset, per year (with 168 hours in a week, we aggregate across each *week-hour* in a given year). This yields 30M sequences of mappers where each sequence describes a list of mappers sharing similar temporal editing patterns, editing tools, and types of edits.

To detect vandalism, we train a Gradient Boosting Decision Tree (GBDT) model with xgboost library [5], which consists of context and content features, similar to [6]. We applied OSM user embeddings into this model by creating two embedding features, `kmeans_cluster` and `cos_sim_last_5_users`. To create `kmeans_cluster`, we ran k-means clustering on OSM users, assigned a cluster to any user with an embedding, and then encoded the cluster based on the average number of edited changesets among each cluster. The idea behind `cos_sim_last_5_users` is that users who are similar to each other are more likely to edit the same objects. Starting with an edit to an OSM object, we compute the cosine similarity between the user responsible for the edit and the previous five mappers that edited the object.

Next, we trained a new model by injecting the embedding features, and we have seen a solid relative improvement of 1.3% in our primary metric, area under the receiver-operator curve (AUC-ROC), which is a widely used metric in vandalism detection research [6]. The feature importance of `kmeans_cluster` is ranked as high as 2/49, with a coverage of 99.9%, while `cos_sim_last_5_users` has an importance rank of 16/49, largely due to a relatively low coverage of 64%, meaning that the majority of edits in OSM create new objects, so there can be no editing history for these.

Because of the AUC improvements and high feature importance, Facebook has deployed this model in production to detect vandalism, as a part of the data validation in the Facebook Map and Daylight Map, a validated, vandalism-free distribution of OSM [7].

The accurately labeled dataset of vandalism to named elements in OSM is a tremendous asset to researchers hoping to further the work of automated vandalism detection. As part of this work, we are publishing this fully labeled vandalism corpus for others in the OSM research community to use [8].

In addition to vandalism detection, OSM user embeddings can be used for automated community detection within OSM. OSM is comprised of many distinct groups of mappers; considering each of these groups a different sub-community makes OSM a "community of communities" [9]. The creation of the temporal and semantic editing patterns were specifically designed to create sequences of mappers with high likelihood of belonging to

the same community, the resulting user embeddings then group these communities of mappers appropriately.

To apply the user embeddings to community detection, we constructed a network graph of OSM users in which two users are connected if they have a high cosine similarity as defined by the user embedding. For each mapper in our embedding, we compute the cosine similarity for the top 100 most similar other mappers. If the cosine similarity is above 0.95, then an edge is created between the two nodes with a weight equal to the actual cosine similarity value. Figure 1 shows a depiction of this network graph for the top 10,000 most prolific editors in 2020.

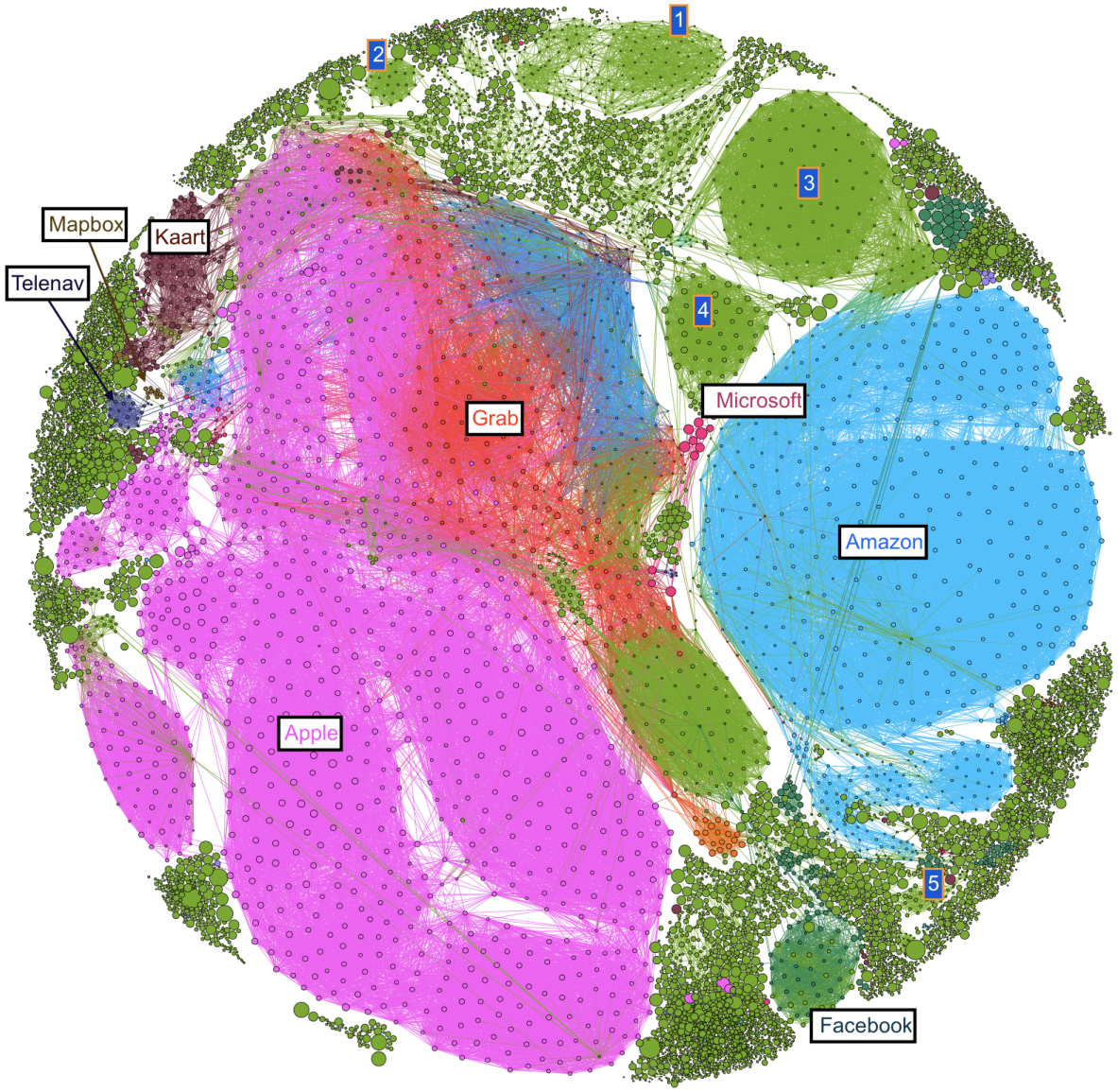


*Figure 1*. Network graph of the top 10,000 mappers in 2020 by changeset count. Nodes (mappers) are colored with regard to corporations they are known to be associated with. Green nodes are not known corporate editors.

Previous research on corporate editing in OSM has identified specific teams of paid mappers working for different corporations [10]. These paid mappers represent one type of editing "community" in OSM. Since these mappers publicly disclose their affiliations, the

members of this mapping community are known. Using the list of paid editors identified in [11], we colored the nodes in Figure 1 with their corporate association, labeled as such. The green nodes represent users that are not known corporate editors. The purpose of this labeling is not to identify corporate editors, but rather to validate the performance of our approach at identifying known communities: The obvious clusters between different colors show that these different corporate communities are successfully identified by the OSM User Embeddings.

Of interest, then, are the additional clusters appearing among the green nodes which may represent specific editing communities in OSM. For an illustrative example, we label 5 example community clusters in Figure 1. Manually investigating the mapping activity of the users associated in each numbered cluster reveals that the network graph successfully clusters other sub communities of editors in OSM. Cluster 1, for example, appears to be a group of Russian editors who joined OSM within the past two years and are heavily focused on editing buildings this past year in small cities near the Caspian Sea. Cluster 2 also appears to have many Russian mappers who joined OSM in the past four years and specifically clean up tags all over the world. Cluster 3 is a group of previously unidentified Amazon mappers, showing the abilities of this approach to potentially help identify more corporate editors. Clusters 4 and 5 represent two distinct groups of humanitarian mappers participating in Humanitarian OpenStreetMap Team (HOT) mapping tasks last year.

## References

[1] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint*, arXiv:1301.3781.

[2] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *arXiv preprint*, arXiv:1310.4546.

[3] Truong, Q. T., Touya, G., & Runz, C. (2020). Osmwatchman: Learning how to detect vandalized contributions in osm using a random forest classifier. *ISPRS International Journal of Geo-Information*, 9(9), 504.

[4] Zaveri, M. (2018). New York City is briefly labeled 'Jewtropolis' on Snapchat and other apps. Retrieved from https://www.nytimes.com/2018/08/30/business/jewtropolis-map-new-york-snapchat.html

[5] Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785-794.

[6] Heindorf, S., Potthast, M., Stein, B., & Engels, G. (2016). Vandalism detection in wikidata. In: *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 327-336.

[7] Facebook (2021). Daylight Map Distribution. Retrieved from https://daylightmap.org

[8] Facebook (2021). OSM Name Vandalism Corpus Released. Retrieved from https://daylightmap.org/2021/05/24/name-vandalism-corpus-release.html

[9] Solís, P., Anderson, J., & Rajagopalan, S. (2020). Open geospatial tools for humanitarian data creation, analysis, and learning through the global lens of youthmappers. *Journal of Geographical Systems*.

[10] Anderson, J., Sarkar, D., & Palen, L. (2019). Corporate editors in the evolving landscape of OpenStreetMap. *ISPRS International Journal of Geo-Information,* 8(5), 232.

[11] Anderson, J. (2021). A 2021 update on paid editing in OpenStreetMap. Retrieved from https://www.openstreetmap.org/user/Jennings%20Anderson/diary/396271