Knowledge-Enriched Route Computation

Georgios Skoumas[1](✉), Klaus Arthur Schmid[2], Gregor Jossé[2], Matthias Schubert[2], Mario A. Nascimento[3], Andreas Züfle[2], Matthias Renz[2], and Dieter Pfoser[4]

[1] National Technical University of Athens, Athens, Greece
gskoumas@dblab.ece.ntua.gr
[2] Ludwig-Maximilians-Universität München, Munich, Germany
{schmid,josse,schubert,zuefle,renz}@dbs.ifi.lmu.de
[3] University of Alberta, Edmonton, Canada
nascimento@ualberta.ca
[4] George Mason University, Fairfax, USA
dpfoser@gmu.edu

Abstract. Directions and paths, as commonly provided by navigation systems, are usually derived considering absolute metrics, e.g., finding the shortest or the fastest path within an underlying road network. With the aid of Volunteered Geographic Information (VGI), i.e., geo-spatial information contained in user generated content, we aim at obtaining paths that do not only minimize distance but also lead through more popular areas. Based on the importance of landmarks in Geographic Information Science and in human cognition, we extract a certain kind of VGI, namely spatial relations that define *closeness* (*nearby, next to*) between pairs of *points of interest* (POIs), and quantify them following a probabilistic framework. Subsequently, using Bayesian inference we obtain a crowd-based *closeness* confidence score between pairs of POIs. We apply this measure to the corresponding road network based on an altered cost function which does not exclusively rely on distance but also takes crowdsourced geo-spatial information into account. Finally, we propose two routing algorithms on the enriched road network. To evaluate our approach, we use Flickr photo data as a ground truth for popularity. Our experimental results – based on real world datasets – show that the paths computed w.r.t. our alternative cost function yield competitive solutions in terms of path length while also providing more "popular" paths, making routing easier and more informative for the user.

1 Introduction

User generated content has benefited many scientific disciplines by providing a wealth of new data. Technological progress, especially smartphones and GPS receivers, has facilitated contributing to the plethora of available information. OpenStreetMap[1] constitutes the standard example and reference in the area of VGI. Authoring geo-spatial information typically implies coordinate-based,

[1] https://www.openstreetmap.org/.

C. Claramunt et al. (Eds.): SSTD 2015, LNCS 9239, pp. 157–176, 2015.
DOI: 10.1007/978-3-319-22363-6_9

quantitative data. Contributing quantitative data requires specialized applications (often part of social media platforms) and/or specialized knowledge, as is the case of OpenStreetMap (OSM).

The broad mass of users contributing content, however, are much more comfortable using *qualitative information*. People typically do not use geo-coordinates to describe their spatial motion, for instance when traveling or roaming. Instead, they use qualitative information in the form of toponyms (landmarks) and spatial relationships ("near", "next to", "close by", etc.). Hence, there is an abundance of geo-spatial information (freely) available on the Internet, e.g., in travel blogs, largely unused. In contrast to quantitative information, which is mathematically measurable, qualitative information is based on personal cognition. Therefore, accumulated and processed qualitative information may better represent the human way of thinking.

This is of particular interest when considering the "routing problem" (equivalent to "path computation"). Traditional routing queries use directions from systems that only take inherent cost measure of the underlying road network into account, e.g., distance or travel time. In human interaction, such information is usually enhanced with qualitative information (e.g. "the street next to the church", "the bridge North of the Eiffel tower"). Combining traditional routing algorithms with crowdsourced geo-spatial references we aim to more properly represent human perception while keeping it mathematically measurable.

In [1], the authors analyze the important role of landmarks for the representation of geographic space in human mind, i.e., people tend to describe their position in space based on landmarks and relations between them. Based on this fact, in this work, we enrich a road network with information about spatial relations between pairs of Points of Interest (POIs) extracted from user generated data (travel blog data). Using these relations, we obtain routes that are easier to interpret and follow, possibly rather resembling a route that a person would provide.

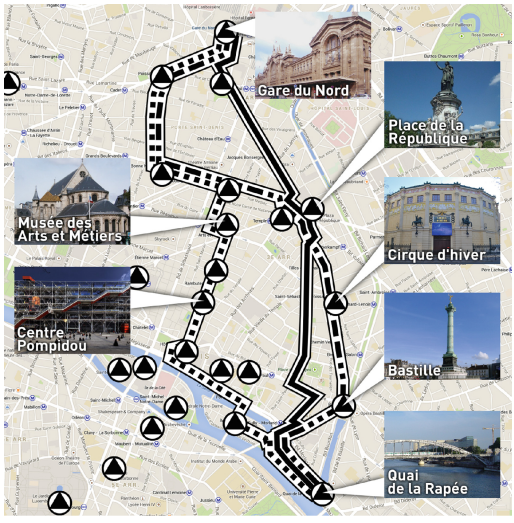


Fig. 1. Shortest (continuous) and alternative paths (dot dashed and dotted) alongside POIs in the city of Paris. This result is an output of some of the algorithms presented in this paper.

As an example, consider the routing scenario in Fig. 1 which is set in the city of Paris, France. The continuous line represents the conventional shortest path from starting point "Gare du Nord" to the target at "Quai de la Rapée" while the dot dashed and dotted lines represent alternative paths computed by the algo-

rithms introduced in this paper. The triangles in this example denote touristic landmarks and sights. For instance, the dot dashed path on the bottom right passing through recognizable locations such as "Place de la République", "Cirque d'hiver" and "la Bastille", as proposed by our algorithms, is considerably easier to describe and follow, and might yield more interesting sights for tourists than the shortest path.

The major challenge in this contribution is the extraction of crowdsourced geo-spatial information from textual data and the enrichment of an existing road network with this information. The enriched road network is subsequently used to provide paths between a given start and target that satisfy the claim of higher popularity (which is formally introduced in Sect. 3), while only incurring a minor additional spatial distance. In addition to this main application, we note that our techniques can furthermore be used to automatically provide interesting touristic routes in any place where information about POIs is available. The transition from textual information to routing in networks is not at all straightforward, therefore we employ and develop various methods from different angles of computing science. In a pre-processing step, we first mine VGI from user generated texts, by employing Natural Language Processing (NLP) methods in order to determine spatial entities (POIs) and spatial relations between them (see Sect. 2). Furthermore, due to the inherent uncertainty of crowdsourced data, we employ probability distributions to quantitatively model spatial relations mined from the text (see Sect. 2.2). Having this information available, we propose and approach for "popular" path computation. To summarize, our contributions are as follows:

- We introduce a Bayesian inference-based transition from the modeled spatial relations to spatial *closeness* confidence measurements according to the crowd (see Sect. 3.1).
- We define a new cost criterion which is used to enrich an underlying road network with the aforementioned confidence measurements (see Sect. 3.2).
- We extend our previously presented road network enrichment approach (see [2]) with a skyline-based road network enrichment approach.
- Finally, we propose two algorithms which use the enriched road network to compute actual paths (see Sect. 4).

2 Pre-processing: Spatial Relation Extraction and Modeling

This section highlights our approach on qualitative data extraction from texts and presents a probabilistic approach for representing spatial relationships based on distance and orientation features. Key ingredients of our approach are NLP methods for information extraction from texts and algorithms that train probabilistic models, which are required due to the inherent uncertainty of crowdsourced data. Our discussion below includes a short description of NLP tools we use to extract spatial relations between POIs, the features we used to model spatial relations as probability distributions, and a short analysis of the modeling

approach used in [3]. These models are necessary to assess the quality of spatial relations extracted from text which will be used in Sect. 3.2 for the enrichment of the underlying road network.

2.1 Spatial Relation Extraction from Texts

In this work, we choose travel blogs as a rich source for (crowdsourced) geospatial data. This selection is based on the fact that people tend to describe their experiences in relation to their trips and places they have visited, which results in "spatial" narratives. To gather such data, we use classical Web crawling techniques and compile a database consisting of 250,000 texts, obtained from 20 travel blogs.

Obtaining qualitative spatial relations from text involves the detection of (*i*) POIs (or toponyms) and (*ii*) spatial relationships linking the POIs. The employed approach involves geoparsing, i.e., the detection of candidate phrases, and geocoding, i.e., linking the phrases to actual coordinate information.

For the relation extraction task we follow the approach used in [4] where a Natural Language Processing Toolkit (NLTK) (cf. [5]) based spatial relation extraction approach is presented. NLTK is a leading platform for analyzing raw natural language data. The search for spatial relations in texts results into triplets of the form (P_i, R^k, P_j), where p_i and p_j are named entities (landmarks) and R^k is the spatial relation that intervenes between P_i and P_j. Following this path, we managed to extract 500,000 POIs from the aforementioned travel blog text corpus. For the geocoding of the POIs, we rely on the GeoNames[2] geographical gazetteer data, which contains over 10 million POI names worldwide and their coordinates. This procedure associates (whenever possible) POIs extracted from travel blogs with geographical coordinates. Using the GeoNames gazetteer we were able to geocode about 480,000 out of the 500,000 extracted POIs and to end up with about 600,000 triplets of the form (P_i, R^k, P_j) worldwide.

For our experiments we want to focus on regions with high triplet density in order to get meaningful results. Therefore, we focus on the cities of Paris and New York. The triplets we extracted for each of these two cities define what we call *Spatial Relationship Graph*, i.e., a spatial graph in which nodes represent POIs and edges are spatial relationships between them. Let us point out that for the scope of this work, i.e., a combination of short and enriched routes, we only consider distance and topological relations that denote closeness (near, close, next to, at, in etc.). The use of relations that denote direction, e.g., North, South etc., or remoteness, e.g., away from, far from etc., is an open direction for future work.

2.2 Modeling Spatial Relations

Feature Extraction In order to train probabilistic models, we need informative features. We model each spatial relation in terms of *distance* and *orientation* as

[2] http://www.geonames.org/.

presented in [3]. Therefore, we extract occurrences of a spatial relation (such as "near") from travel blogs. For each occurrence, we create a two-dimensional spatial feature vector $D = (D_d, D_o)^\intercal$ where D_d denotes the distance and D_o denotes the orientation between P_i and P_j. Specifically, assuming a projected (Cartesian) coordinate system, the distance between two POIs P_i and P_j is computed as the Euclidean metric between the two respective coordinates. The orientation is established as the counterclockwise rotation of the x-axis, centered at point P_j, to point P_i. This way, we end up with a set of two-dimensional feature vectors $\mathcal{D}_{\text{rel}} = \{D_1, D_2, \ldots, D_n\}$ for each spatial relation. We will use the set of two-dimensional feature vectors in order to train a probabilistic model for each spatial relation.

Probabilistic Modeling. As described in [3], by using a set of two-dimensional feature vectors for each spatial relation such as "near" or "into", we can train Gaussian Mixture Models (GMMs), which have been extensively used in many classification and general machine learning problems [6].

In general, a GMM is a weighted sum of M-component Gaussian densities as $p(d|\lambda) = \sum_{i=1}^{M} w_i g(d; \mu_i, \Sigma_i)$ where d is a l-dimensional data vector (in our case $l = 2$), w_i are the mixture weights, and $g(d; \mu_i, \Sigma_i)$ is a Gaussian density function with mean vector $\mu_i \in \mathbb{R}^l$ and covariance matrix $\Sigma_i \in \mathbb{R}^{l \times l}$. To fully characterize the probability density function $p(d|\lambda)$, one requires the mean vectors, the covariance matrices and the mixture weights. These parameters are collectively represented as $\lambda = \{w_i, \mu_i, \Sigma_i\}$ for $i = 1, \ldots, M$.

Let $\mathcal{R} = \{R^1, \ldots, R^n\}$ denote the set of all spatial relations that we take into account. In our setting, each relation R^k is modeled under a probabilistic framework by a 2-dimensional GMM, trained on each relation's set of two-dimensional feature vectors $\mathcal{D}_{\text{rel}}$. For the parameter estimation of each GMM, we use Expectation Maximization (EM) [7]. EM enables us to update the parameters of a given M-component mixture with respect to a feature vector set $\mathcal{D}_{\text{rel}} = \{D_1, \ldots, D_m\}$ with $1 \leq j \leq m$ and $D_j \in \mathbb{R}^l$, such that the log-likelihood $\mathcal{L} = \sum_{j=1}^{m} \log(p(D_j|\lambda))$ increases with each re-estimation step, i.e., EM re-estimates model parameters λ until convergence. Further details on modeling spatial relations under a probabilistic framework are given in [3].

This procedure results in a trained GMM of the form $p_k(D|\lambda)$, for each spatial relation $R^k, 1 \leq k \leq n$. Given a distance and orientation vector, we can use this model to estimate the probability that a particular relation exists. Based on this information, by bayesian inference we derive a closeness score for pairs of POIs. This procedure is described in the next section.

3 Road Network Enrichment

In this section, we describe our approach to enrich an actual road network with crowdsourced geo-spatial information. Our discussion below includes a description of how we transform a *Spatial Relationship Graph*, as presented in Sect. 2.1, into a weighted graph, and how we use the edge weights of the weighted graph in order to modify the edge costs of a real road network.

3.1 From Relationship to Weighted Graphs

As presented in Sect. 2, the spatial relation extraction procedure results in a relationship graph between POIs. A simple example of such a graph is shown in Fig. 2. In general, let $\mathcal{P} = \{P_1, \ldots, P_m\}$ denote the set of nodes representing the POIs, and let $\mathcal{R} = \{R^1, \ldots, R^n\}$ denote the pre-defined set of spatial closeness relations, represented by spatial NLP expressions like "next to" or "close by".

Furthermore, let $R_{i,j} \subseteq \mathcal{R}$ denote the set of relations extracted from the text between two distinct nodes P_i and P_j. Note that R^k denotes an abstract relation, while $R_{i,j}$ denotes a set of occurrences of relations between a pair of nodes. Let $D_{i,j}$ denote the spatial feature vector (distance and orientation), between two distinct POIs P_i and P_j (as presented in Sect. 2.2). Finally, let $\mathcal{D} := \bigcup_{i \neq j \wedge R_{i,j} \neq \emptyset} D_{i,j}$ denote the set of all spatial feature vectors between all pairs of POIs which have non-empty sets of relations.

We want to estimate the posterior probability of a class $R^k \in R_{i,j}$ based on the spatial feature data $D_{i,j}$ between two POIs P_i and P_j. This is given by Eq. 1. Here, $p(D_{i,j}|R^k)$ denotes the likelihood of $D_{i,j}$ given relation R^k based on the trained GMM (presented as $p(D|\lambda)$ Sect. 2.2), while $P(R^k)$ denotes the prior probability of relation R^k given only the observed relations $R_{i,j}$.

$$P(R^k|D_{i,j}) = \frac{p(D_{i,j}|R^k)P(R^k)}{\sum_{l=1}^{n} p(D_{i,j}|R^l)P(R^l)} \tag{1}$$

In a traditional classification problem the spatial relation R^k between a pair of POIs would be classified to the spatial relation model with the highest posterior. In contrast to this approach, we consider each posterior probability $P(R^k|D_{i,j})$ as a measure of confidence of the existence of relation R^k between P_i and P_j. Remember that all the relations we consider reflect terms of spatial closeness.

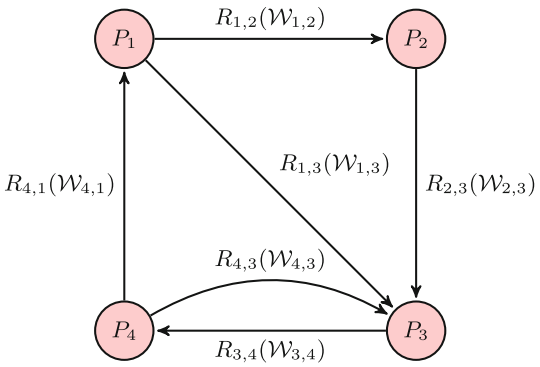


Fig. 2. Simple relationship graph. Nodes represent POIs and each edge represents the set of relations $R_{i,j}$ through which its adjacent nodes P_i and P_j are connected. Each of these sets is mapped onto the closeness score $\mathcal{W}_{i,j}$, turning the relationship into a weighted graph.

We combine all these posteriors into one measure which we refer to as *closeness score* $\mathcal{W}_{i,j}$ of the pair of POIs P_i and P_j, defined in Eq. 2.

$$\mathcal{W}_{i,j} = \frac{1}{|\mathcal{R}|} \cdot \sum_{i=1}^{|R_{i,j}|} \frac{P(R^k|D_{i,j})}{\max_k\{P(R^k|\mathcal{D})\}} \tag{2}$$

Here, we sum all the posteriors $P(R^k|D_{i,j})$ normalized by the maximum posterior of each relation in the relationship graph and we normalize the summation by the total number of spatial relations in the relationship graph. This is done for all pairs P_i, P_j where $R_{i,j} \neq \emptyset$. We refer to these pairs as *close* since at least one of our relations, reflecting closeness, exists. As is illustrated in Fig. 2, assigning the respective weights $\mathcal{W}_{i,j}$ to the edges of the relationship graph, we obtain a weighted graph. Note that $\mathcal{W}_{i,j} \in [0, 1]$ but typically $0 < \mathcal{W}_{i,j} \ll 1$. In Sect. 5 the influence of $\mathcal{W}_{i,j}$ on the results is examined, in particular, different scalings are tested. In this weighted relationship graph, denoted by H^*, there exists a vertex for each POI and an edge (P_i, P_j) (equipped with weights $\mathcal{W}_{i,j}$ and Euclidean distances d_{ij}) for each pair of POIs P_i, P_j that are close in the above sense ($R_{i,j} \neq \emptyset$).

3.2 From Weighted Graphs to Road Network Enrichment

Now that we have extracted and statistically condensed the crowdsourced data into a closeness score, we need to apply the obtained closeness scores to the underlying network. We have investigated several strategies and have decided upon a compromise between simplicity and effectiveness. We will present two road network enrichment approaches and we propose two algorithms on routing with enriched graphs. The first enrichment approach, also analyzed in our previous work in [2], is based on Djikstra shortest path computation while the second is based on Skyline path computation.

Initially, let $G = (V, E, d)$ denote the graph representing the underlying road network, i.e., the vertices $v \in V$ correspond to crossroads, dead ends, etc., the edges $e \in E = V \times V$ represent roads connecting vertices. Furthermore, let $d : E \to \mathbb{R}_0^+$ denote the function which maps every edge onto its distance. We assume that $\mathcal{P} \subseteq V$, i.e., each POI is also a vertex in the graph. This is only a minor constraint since we can easily map each POI to each nearest node on the graph or introduce pseudo-nodes. Our two enrichment methods are described below.

Djikstra Shortest Path Approach. For each pair of spatially connected POIs, P_i, P_j, we compute the shortest path connecting P_i and P_j in G, which we denote by $r(i, j)$. We then define a new cost function $c : E \to \mathbb{R}_0^+$ which modifies the previous cost $d(e)$ of an edge as follows:

$$c(e) = d(e) \cdot \prod_{e \in r(i,j)} (1 - \alpha \mathcal{W}_{i,j}) \tag{3}$$

where $e \in r(i,j)$ iff e is an edge within the shortest path from P_i to P_j and where $\alpha \in [0,1]$ is a weight scaling factor to control the balance between the spatial distance $d(e)$ and the modification caused by the closeness score $\mathcal{W}_{i,j}$. In the case of $\alpha = 0$, we obtain the unadapted edge weight $c(e) = d(e)$. Summarizing, the more shortest paths between POI pairs run through e, the lower its adjusted cost $c(e)$. The reason for enriching the shortest paths is that they represent the most intuitive connections between any two points in a road network.

We now define the *enriched graph* $G^* = (V, E, c)$. It consists of the original vertices and edges and is equipped with the new cost function which implies the re-weighting of edges. Any path computation algorithm in G^* (e.g. a Dijkstra search) therefore favors edges which are part of shortest paths between POIs which are close according to the crowd. When computing the cost of a path on G^*, as before, we sum the respective edge weights which now differ from the original edge weights (due to the altered cost function). We refer to this procedure of incorporating the crowdsourced information as *D-enrich*.

Path Skyline Approach. One shortcoming of *D-enrich* is the assumption that the crowd unanimously favors exactly one path to connect a pair of POIs P_i and P_j, namely the shortest path. Especially in multicriteria networks which comprise of a set of cost criteria, e.g., travel time, energy consumption, road tolls, optimality is usually defined as a personal trade-off between the given criteria. For example: How much additional time has to be spent to avoid a toll road? However, defining this trade-off numerically as a vector of preferences is not reasonable, and even if it would be, finding the personally preferred trade-offs for all users is in general not possible. Therefore, the best practice is to present a set of alternative paths to the user. The most established and very comprehensive set of alternative paths is the so-called path skyline [8]. This set contains all paths which are non-dominated in the following sense: The cost vector u dominates a cost vector v, denoted $u \prec_{\text{dom}} v$, if u has a smaller cost value than v in at least one dimension i and v does not have a smaller cost value than u in any dimension j. Hence, the path skyline comprises all paths which are optimal under some monotone combination function of the cost criteria. Hence, the path skyline contains all optimal paths for all possible trade-offs between the cost criteria.

To enrich our road network, we compute the path skyline (w.r.t. distance and travel time) as proposed in [9] between each pair of spatially connected POIs P_i and P_j in G, denoted by $s(i,j)$. Although the paths contained in $s(i,j)$ differ from one another, they often share some edges. Simply following each path for enrichment might unnecessarily favor edges contained in many skyline paths. Therefore, we adjust the weights of edges independent of the number of skyline paths in which they occur. Let $S_{i,j} \subset E$ denote the set of all distinct edges which are part of at least one skyline path from P_i to P_j. Analogously to *D-enrich*, we define the cost function $c : E \to \mathbb{R}_0^+$ to modify the original cost $d(e)$ of an edge, as before. While the adjusted cost function is the same as before (see Eq. 3), the set of edges with adjusted costs is a superset, i.e., $S_{i,j} \supseteq r(i,j)$.

We now define the *enriched graph* $G^{**} = (V, E, c)$. It consists of the original vertices and edges equipped with the altered cost function reflecting a reweighting of edges contained in skyline paths. Any path computation algorithm in G^{**} (e.g. a Dijkstra search) therefore favors edges which are part of the Skyline paths between POIs which are close according to the crowd. We refer to this procedure of incorporating the crowdsourced information as *S-enrich*.

3.3 Influence of Adjusted Costs

In order to measure the influence of the adjusted cost values along a computed path $p = (e_1, \ldots, e_r)$ on an enriched graph (G^*or G^{**}), we introduce the *enrichment ratio* (ER) function er.

$$er(p) = \frac{1}{d(p)} \sum_{i=1}^{r} c(e_i) \tag{4}$$

Here, $d(\cdot)$ and $c(\cdot)$ are as in the previous two sections. By normalizing with the total length of the path, we are able to compare the spatial connectivity of paths independent of length as well as start and target nodes. Here, a lower ratio implies higher closeness score values along the edges of the path. If none of the edges of a path is part of any shortest or skyline path between POIs, its enrichment ratio is 1, while the (highly unlikely) optimal enrichment ratio is 0. On the enriched graphs G^* and G^{**} we may now define our path computation algorithms.

4 Path Computation on Enriched Graphs

Now that we have a measure quantifying the enrichment of a path, we investigate the effect of *D-enrich* and *S-enrich* on the actual path computation. For this purpose, we present two approaches which make use of the enriched network and the weighted relationship graph H^* (Sect. 3.1). In Sect. 5 they are compared to the conventional shortest paths within the original graph, as obtained with Dijkstra's algorithm, which we denote by Dij-G.

Note that for the evaluation procedure, all paths in this paper are computed by Dijkstra's algorithm because our main focus is not the routing itself but the incorporation of textual information into existing road networks. If desired, speed-up techniques, such as preprocessing steps and/or other search algorithms, could easily be employed.

Our first approach, given start and target nodes, executes a Dijkstra search in the enriched road network graph G^* or G^{**} w.r.t. the adjusted cost function. Depending on the enrichment used, *D-enrich* or *S-enrich*, we refer to the first algorithm as Dij-G*or Dij-G**, respectively.

Our second approach, uses the enriched road network graphs G^* or G^{**} as well as the weighted relationship graph H^*. Given start and target nodes within the enriched graph (G^*or G^{**}), entry and exit nodes within H^* are determined.

Subsequently, we route within H^*, i.e., from POI to POI, again using Dijkstra's algorithm. Depending on the enrichment used, *D-enrich* or *S-enrich*, we refer to the second approach we want to present as Dij-H* or Dij-H**, respectively. Note that in both cases we use the same graph H^*, but we refer to the *S-enrich* case as Dij-H** in order to differentiate the two methods.

All approaches, return paths connecting start and target. But while Dij-G computes the shortest path in the original graph G, all the approaches compute the shortest paths in the enriched graphs w.r.t. the adjusted cost function c. By construction of c, it favors edges which are part of the Dijkstra shortest paths or the skyline paths, between close POIs. Dij-H* and Dij-H** in contrast, do not only favor these edges, but are restricted to them. Having found entry and exit nodes within H^*, Dij-H* and Dij-H** hop from POI to POI in direction of the target. Hence, Dij-G, Dij-G*, Dij-G**, Dij-H*, Dij-H** in that order, represent an increasing binding to the extracted relations. Dij-G is not bound to the relations at all, while Dij-G* and Dij-G** (by the adjusted cost function) favors "relation-edges", and Dij-H* and Dij-H** are strictly bound to the relations and the graph formed by them.

Let us formalize Dij-H* (Dij-H** can be formalized in the same way). Given start and target node in G^* (or G^{**} for the Dij-H** case), it first determines the so-called entry and exit nodes to and from H^*. However, to exclude POIs which would imply a significant detour, we restrict the set of valid POIs, i.e., we restrict the search to a subgraph of H^*, denoted as h^*. Figure 3 illustrates our computationally inexpensive implementation of a query ellipse that allows for some deviation in the middle of the path as well as for minor initial and final detours.

The pseudo-code for the second approach is given in Algorithm 1. Here, we present only the Dij-H* case, since Dij-H** works in the same way by utilizing the G^{**} graph. After selecting the valid set of POIs (Step 2), entry and exit nodes to and from H^* are determined, i.e., the closest POIs to start and target node, respectively (Steps 4 and 5). Entry and exist nodes connect the road network G^* to the relationship graph H^*. Subsequently, the shortest path in h^* from entry to exit node is computed using Dijkstra's algorithm w.r.t. the Euclidean distance (Step 5). Note that a shortest path within H^* is a sequence of POIs. We therefore map this sequence onto G^* by computing the shortest paths between the consecutive pairs of POIs in G^* w.r.t. the adjusted cost function (Step 8). Also, we

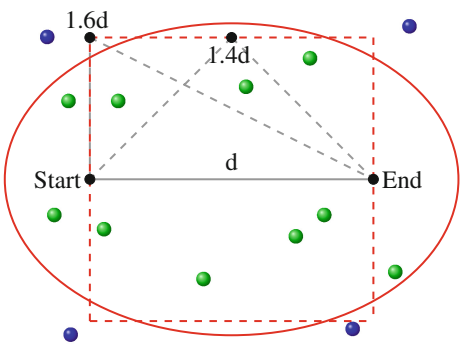


Fig. 3. Restriction of relationship graph H^* to a subgraph h^*, in order to avoid implausible detours. The green dots represent POIs, i.e., nodes of H^* which are also in h^*, the blue ones are left out (color figure online).

Algorithm 1. Dij-H*

Input: Enriched Graph G^*, Spatial Relationship Graph H^*, start s, target t
Output: Path p between s and t

```
1  begin
2      h* ← subgraph of H* in bounding ellipse
3      p ← empty path
4      P_entry ← select POI P ∈ h* closest to s
5      P_exit ← select POI P ∈ h* closest to t
6      p_h ← Dijkstra(h*, P_entry, P_exit)
7      predecessor ← s
8      foreach POI P on path p_h do
9          v ← select node v ∈ G* representing P
10         p.APPEND(Dijkstra(G*, predecessor, v))
11         predecessor ← v
12     end
13     p.APPEND(Dijkstra(G*, last, t))
14     return p
15 end
```

compute the shortest paths in G^* from start to entry node and exit to target node. Concatenating these paths (start to entry, POI to POI, exit to target), we return a full path.

5 Experimental Evaluation

In this section, we want to investigate the effect and impact of the network enrichment. We compare the results of the conventional Dijkstra search, Dij-G, to the results of Dij-G* and Dij-H*, which use the Djikstra shortest path enriched (*D-enrich*) graph G^*, and the results of Dij-G** and Dij-H**, which use the skyline path enriched (*S-enrich*) graph G^{**}. All approaches are evaluated on real world datasets. Besides comparing the computed path w.r.t. their enrichment ratio (ER) and length (as presented in Sect. 3.2), we introduce a measure of popularity based on Flickr data, which is explained in the following section. All the text processing parts were implemented in Python while modeling parts were implemented in Matlab. Network enrichment and path computation tasks were conducted using the Java-based MARiO Framework [10] on an Intel(R) Core(TM) i7-3770 CPU at 3.40 GHz and 32 GB RAM running Linux (64 bit).

5.1 Enrichment Ratio, Distance and Popularity Evaluation

Our experiments are set in two cities, Paris and New York. These regions have comparatively high density of spatial relations, Flickr photo data, and OSM data, which accounts for an exact representation of the road networks. As mentioned before, we compare the output of Dij-G, Dij-G*, Dij-H*, Dij-G** and

Table 1. Statistics for the weighted relationship graphs, Flickr datasets and road networks of Paris and New York respectively.

	Relationship Graph (H^*)		Flickr		Road Network (G)	
Dataset	# POI Pairs	# Relations	# Photos	# Max Photos per Vertex	# Vertices	# Edges
Paris	400	$2K$	$400K$	100	$550K$	$300K$
New York	300	$1.5K$	$90K$	200	$220K$	$120K$

Dij-H** w.r.t. to the paths they return, more precisely, w.r.t. ER and length of these paths. Since ER is a measure introduced in this paper, we use Flickr data as an independent ground truth. We are aware that to cognitive aspects (like the importance of sights or the value of landmarks) there is no absolute truth. However, in order to be able to draw comparisons, we presume that if the dataset is large enough, the bias can be neglected. We use a geotagged Flickr photo dataset, provided by the authors in [11], to assign a number of photos to each vertex of the underlying road network. The number of Flickr photos assigned to each vertex is referred to *popularity*. In our settings, every photo which is within the 20-meter radius of a vertex, contributes to the popularity of that vertex. The popularity of a path is computed by the summation of all popularity values along this path.

The sizes of the weighted relationship graphs H^*, road network and Flickr photo data for both cities are shown in Table 1. Regarding the weighted relationship graphs, we provide the number of unique POI pairs extracted from the travel blog corpus and the number of spatial (closeness) relations extracted between them, as was presented in Sect. 2. Regarding Flickr data, we provide the total number of geotagged photos in each city and the maximum number of photos assigned to one vertex of the road network. Finally, regarding the road network, we provide the total number of edges and vertices. Note that although the datasets differ in terms of density (w.r.t. to relations and Flickr photos), our algorithms provide similar results.

We present two experimental settings: In Setting (i) we examine the influence of different scalings of the closeness score $\mathcal{W}_{i,j}$ in terms of enrichment ratio, path length increase (distance) and popularity. Setting (ii) investigates the influence of the path length, i.e., the distance between start and target is varied, again in terms of enrichment ratio, path length increase (distance) and popularity. In both settings we present the ER performance of the algorithms separately from their performance in terms of distance and popularity as ER is a measure that mainly proves that our network enrichment approach works properly, i.e., ER should increase with the increase of the influence of $\mathcal{W}_{i,j}$ on the network and the increase of the path length. Hence, based on our own measure (ER) we validate that the proposed approach works properly.

In Setting (i), for 100 randomly chosen pairs of start and target nodes the respective shortest paths within the actual road network are computed using

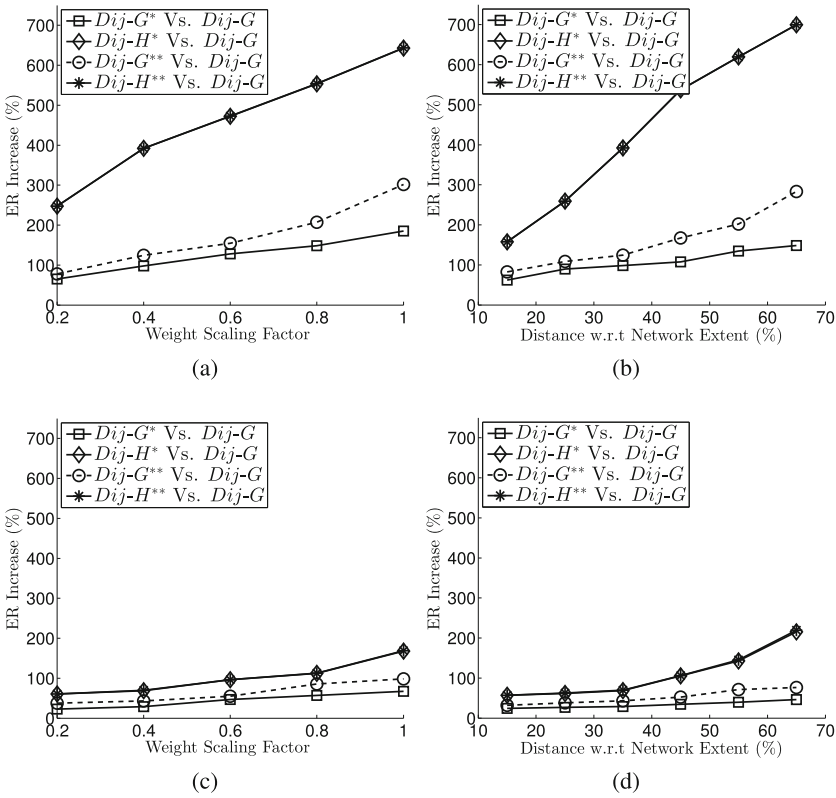


Fig. 4. (a), (b) show ER increase for algorithms Dij-G* and Dij-H* for Paris dataset for Settings *i* and *ii* respectively. (c), (d) show ER increase for algorithms Dij-G* and Dij-H* for New York dataset for Settings *i* and *ii* respectively.

Dijkstra's algorithm, Dij-G. Continuing, for the same start and target pairs, we run Dij-G*, Dij-H*, Dij-G** and Dij-H**. Subsequently, for each pair the difference w.r.t. ER, distance and popularity is computed, and finally averaged over all pairs. We require the distance between start and target nodes to be at least 30 % and at most 50 % of the Euclidean extent of the network (approximately 6 km to 10 km), in order to exclude paths which start and end in the outskirts of the city (where there are few to no POIs). Figure 4 ((a), (c)) show the influence of the closeness score $\mathcal{W}_{i,j}$ on ER for the datasets of Paris and New York respectively. As we increase the impact of $\mathcal{W}_{i,j}$, we observe an increase of ER for all four cases in comparison to Dij-G in both datasets. For the Paris dataset, the increase in ER is in the range of 80 % to 250 % for the Dij-G* and Dij-G**, with the latter performing better, and in the range of 250 % to 620 % for Dij-H* and Dij-H** with the latter performing better. For the New York dataset, the

increase in ER is in the range of 20 % to 80 % for the Dij-G* and Dij-G**, with the latter performing better, and in the range of 80 % to 150 % for Dij-H* and Dij-H**, with the latter performing better.

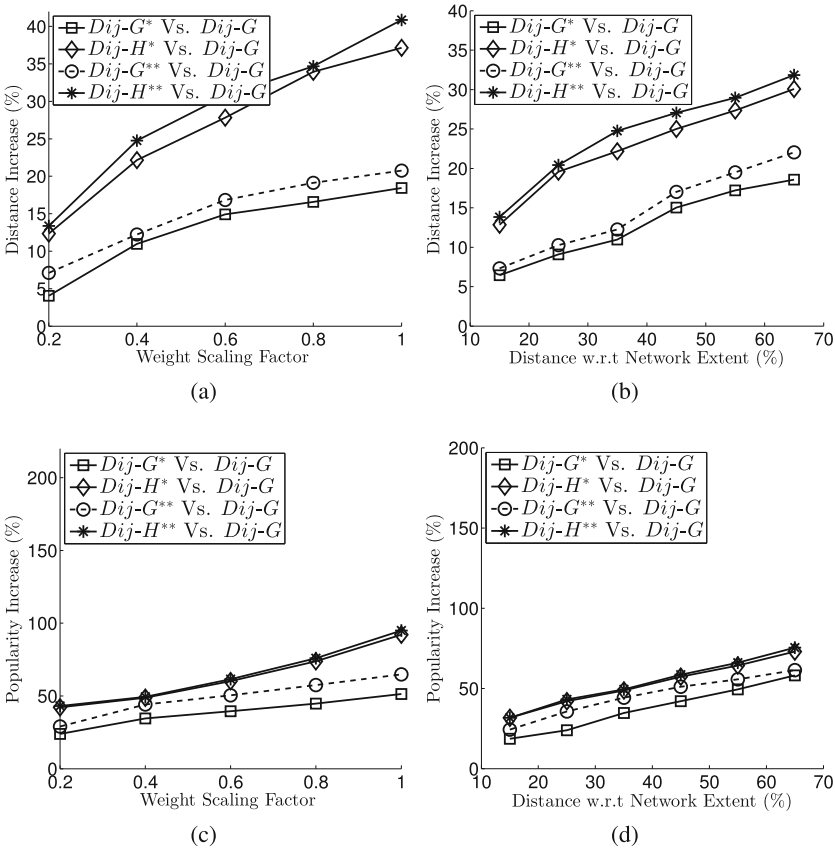


Fig. 5. (a), (c) show Distance and Flickr popularity increase for algorithms Dij-G*and Dij-H*for Paris dataset for experimental Setting *i*. (b), (d) show Distance and Flickr popularity increase for algorithms Dij-G*and Dij-H*for Paris dataset for experimental Setting *ii*.

Moreover, the first column of Figs. 5 and 6 ((a), (c)) shows the influence of weight scaling factor $\mathcal{W}_{i,j}$ on distance and popularity. As we increase $\mathcal{W}_{i,j}$ from 0.2 to 1.0, we observe an increase of distance and popularity for both cases in comparison to Dij-G in both datasets. The increase among all datasets, in terms of path length is in the range of 3 % to 16 % for Dij-G* and Dij-G**, and in the range of 7 % to 38 % for Dij-H* and Dij-H**. Additionally, the increase in popularity is in the range of 30 % to 120 % for Dij-G* and Dij-G**, and in the range of 40 % to 160 % for Dij-H* and Dij-H**.

It is clear that Dij-G* and Dij-G** always perform better than Dij-H* and Dij-H** in terms of path length increase, but Dij-H*and Dij-H**perform always better in terms of ER and popularity. This is because Dij-H*and Dij-H** route directly through the POIs, causing greater detours, but passing along highly weighted parts of the enriched graphs (G^*or G^{**}), which mostly coincide with dense Flickr photo regions. Moreover, it is clear that *S-enrich* always performs better than *D-enrich*, in terms of ER and popularity with a very short increase, of about 2–3 % in path length. This validates that skyline enrichment provides competitive paths in terms of distance (minorincrease) and popularity (significant increase).

Continuing, in Setting (*ii*) we vary the distance between and target, relative to the extent of the whole network. We consider five different distance brackets of shortest paths in the original graph G, the first one ranging from 10 % to 20 %, the last one ranging from 50 % to 60 % of the extent of the whole network. For 100 randomly chosen pairs of start and target nodes (within the respective distance bracket) paths with Dij-G, Dij-G*, Dij-G**, Dij-H* and Dij-H** are computed. As before, for each pair the difference w.r.t. ER, distance and popularity is computed and averaged over all pairs. Figure 4 ((b), (d)) show the increase of ER as we proceed through the distance brackets for both datasets. The second column of Figs. 5 and 6 ((b), (d)) show the results in terms of distance and popularity increase. As we proceed through the distance brackets, we observe an increase of the distance and popularity for all cases in comparison to Dij-G in both datasets. The increase among all datasets, in terms of path length, is in the range of 3 % to 18 % for Dij-G* and Dij-G**, and in the range of 5 % to 30 % for Dij-H* and Dij-H**. Finally, the increase in terms of popularity is in the range of 10 % to 70 % for Dij-G* and Dij-G**, and in the range of 30 % to 140 % for Dij-H* and Dij-H**. As in our previous experimental setting, it is clear that Dij-G* and Dij-H* always perform slightly better (only 2–3 %) in terms of path length increase, while Dij-G** and Dij-H* always outperform Dij-G* and Dij-H* in terms of enrichment ratio and popularity. This underlines the validaty of *S-enrich*, as it provides significantly more popular paths while only incurring minor detours (2–3 % in terms of path length).

Here, we may conclude that both *D-enrich* and *S-enrich* approaches show convincing results. Both cases yield significant increase in terms of ER as well as in terms of the independent Flickr-based measure popularity, while increasing path length only slightly. In the best case, ER increase amounts to almost 700 % while popularity increase amounts to almost 160 % (in comparison to the conventional shortest paths, as computed by Dij-G), while the worst case increase in path length is about 38 % with most cases being less than 10 %. Overall, *D-enrich* works slightly (2–3 %) better in terms of path length while the *S-enrich* is always significantly better (more than 10 % in most of the cases) in terms of popularity scores. Consequently, we can claim that spatial relations, extracted from crowdsourced information, can indeed be used to enrich actual road networks and define an alternative kind of routing which reflects what people perceive as "close".

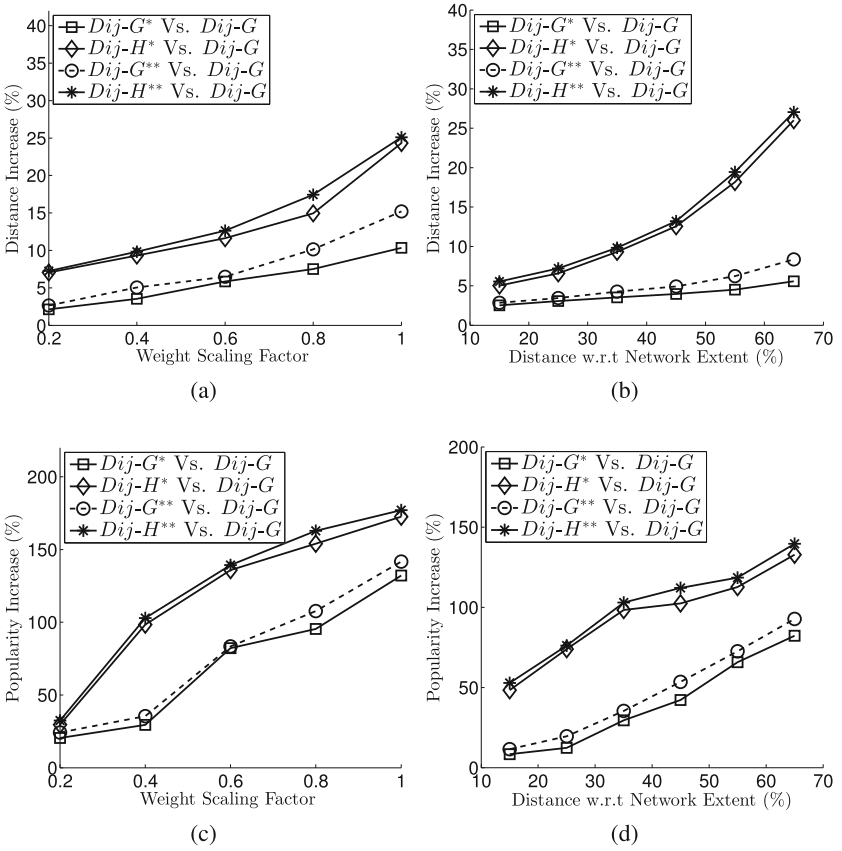
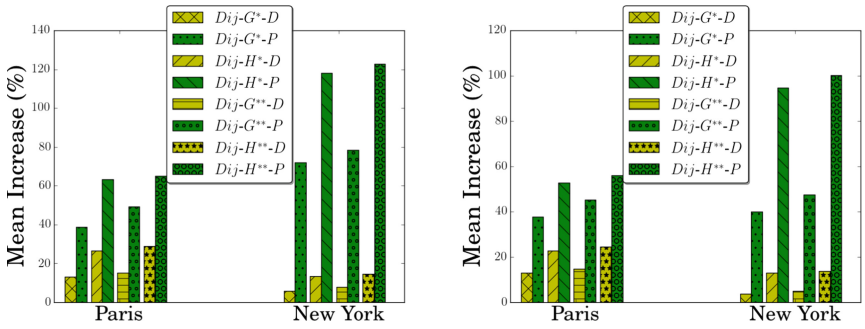


Fig. 6. (a), (c) show Distance and Flickr popularity increase for algorithms Dij-G* and Dij-H* for New York dataset for experimental Setting *i*. (b), (d) show Distance and Flickr popularity increase for algorithms Dij-G* and Dij-H* for New York dataset for experimental Setting *ii*.

Finally, Fig. 7 illustrates the trade-off (mean distance and popularity increase overall experiments) that we take by deviating from the shortest path in order to obtain more interesting paths. This figure shows the relative increase in distance and popularity of the paths returned by our proposed approaches, compared to the baseline approach Dij-G. Here, we use letter D to refer to the distance increase while we use letter P to refer to popularity increase. For both datasets, we can observe that by road network enrichment we can obtain a significant increase in popularity of up to 120 % for the meager price of no more than 25 % additional distance incurred in both experimental settings. With the proposed *S-enrich* approach we achieve to significantly increase popularity while keeping the distance increase almost in the same levels with the *D-enrich* approach.



(a) Setting i (b) Setting ii

Fig. 7. Trade-off between distance and popularity increase of paths

6 Related Work

Research areas relevant to this work include: (*i*) qualitative routing and (*ii*) mining of semantic information from moving object trajectories and trajectory enrichment with extracted semantic information. In what follows, we discuss previous work in both of these areas.

While finding shortest paths in road networks is a thoroughly explored research area, qualitative routing has hardly been explored. Nevertheless, providing meaningful routing directions in road networks is a research topic of great importance. In various real world scenarios, the shortest path may not be the ideal choice for providing directions in written or spoken form, for instance when in an unfamiliar neighborhood, or in cases of emergency. Rather, it is often more preferable to offer "simple" directions that are easy to memorize, explain, understand and follow. However, there exist cases where the simplest route is considerably longer than the shortest. The authors in [12] and [13] try to tackle the problem of efficient routing by using cost functions that trade off between minimizing the length of a provided path while also minimizing the number of turns on the provided path. The major shortcoming of these approaches is that they focus almost exclusively on road network data without taking into account any kind of qualitative information, i.e., information coming from the user. Opposed to that, we try to approach the problem of efficient routing by integrating spatial knowledge coming from the crowd thus enriching an actual road network.

The discovery of semantic places through the analysis of raw trajectory data has been investigated thorougly over the course of the last years. The authors in [14–16] provide solutions for the semantic place recognition problem and categorize the extracted POIs into pre-defined types. Moreover, the concept of "semantic behavior" has recently been introduced. This refers to the use of semantic abstractions of the raw mobility data, including not only geometric patterns but also knowledge extracted jointly from the mobility data as well as the underlying geographic and application domains in order to understand the actual behaviour

of moving users. Several approaches like [17,21] have been introduced the last decade. The core contribution of these articles lies in the development of a semantic approach that progressively transforms the raw mobility data into semantic trajectories enriched with POIs, segmentations and annotations. Finally, recent work [22], can extract and transform the aforementioned semantic information into a text description in the form of a diary. The major drawback of these approaches is that they do not intergrate the extracted semantic information into the road network. Instead, they use the extracted information only on specific trajectories. In our contribution, we analyze crowdsourced data in order to extract semantic spatial information and intergrate it into an actual road network. This will enable us to provide routes that are near-optimal w.r.t. distance while spatially more popular according to the crowd.

7 Conclusions and Outlook

In this work we presented new approaches to computing knowledge-enriched paths within road networks. We incorporated novel methods to extract spatial relations between pairs of POIs, such as "near" or "close by", from crowdsourced textual data, namely travel blogs. We quantified the extracted relations using probabilistic models to handle the inherent uncertainty of user-generated content. Based on these models, we proposed a new cost function to enrich real world road networks, based on Djikstra and skyline path computation. The new cost function reflects the closeness aspect according to the crowd. In contrast to existing approaches, we did not enrich previously computed paths with semantical information, but the entire network. Continuingly, two routing algorithms were presented taking this closeness aspect into account. Finally, we evaluated our ideas on two real world road network datasets, i.e., Paris, France, and New York City, USA. We used metadata from geotagged Flickr photos as a ground truth to support our initial goal of providing more popular paths. All our approaches performed very well by providing slightly longer paths but with significantly higher values of popularity. For future work, we are researching alternative methods for aggregating all categories of spatial relations. Furthermore, we would like to investigate ways to suggest the popular path descriptions to the user based on the POIs they will encounter underway.

Acknowledgements. The research leading to these results has received funding from the EU FP7 project GEOSTREAM (grant No. FP7-SME-2012-315631) as well as the Shared-E-Fleet project by the German Federal Ministry of Economics and Technology (grant No. 01ME12107), the Deutsche Forschungsgemeinschaft (DFG) under grant number RE 266/5-1 and from the DAAD supported by the BMBF under grant number 57052426. Mario A. Nascimento has been partially supported by NSERC Canada. Dieter Pfoser has been partially supported by NGA NURI (grant No. HM02101410004).

References

1. Richter, K.F., Winter, S.: Cognitive aspects: how people perceive, memorize, think and talk about landmarks. In: Landmarks, pp. 41–108. Springer International Publishing, Cham (2014)
2. Skoumas, G., Schmid, K.A., Jossé, G., Züfle, A., Nascimento, M.A., Renz, M., Pfoser, D.: Towards knowledge-enriched path computation. In: Proceedings of the 22nd ACM International Conference on Advances in Geographic Information Systems, 485–488 (2014)
3. Skoumas, G., Pfoser, D., Kyrillidis, A.: On quantifying qualitative geospatial data: a probabilistic approach. In: Proceedings of the Second ACM International Workshop on Crowdsourced and Volunteered Geographic Information, pp. 71–78 (2013)
4. Skoumas, G., Pfoser, D., Kyrillidis, A.T.: Location estimation using crowdsourced geospatial narratives. In: CoRR abs/1408.5894 (2014)
5. Loper, E., Bird, S.: NLTK: The natural language toolkit. In: Proceedings of the ACL 2002 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics , vol. 1, pp. 63–70 (2002)
6. Bishop, C.M.: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York Inc, Secaucus (2006)
7. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. J. Roy. Stat. Soc. Ser. B **39**, 1–38 (1977)
8. Borzsony, S., Kossmann, D., Stocker, K.: The skyline operator. In: Proceedings of the 17th International Conference on Data Engineering, pp. 421–430 (2001)
9. Shekelyan, M., Jossé, G., Schubert, M.: Paretoprep: fast computation of path skylines queries. In: CoRR abs/1410.0205 (2014)
10. Graf, F., Kriegel, H.-P., Renz, M., Schubert, M.: MARiO: multi-attribute routing in open street map. In: Pfoser, D., Tao, Y., Mouratidis, K., Nascimento, M.A., Mokbel, M., Shekhar, S., Huang, Y. (eds.) SSTD 2011. LNCS, vol. 6849, pp. 486–490. Springer, Heidelberg (2011)
11. Mousselly-Sergieh, H., Watzinger, D., Huber, B., Döller, M., Egyed-Zsigmond, E., Kosch, H.: World-wide scale geotagged image dataset for automatic image annotation and reverse geotagging. In: Proceedings of the 5th ACM Multimedia Systems Conference, pp. 47–52 (2014)
12. Sacharidis, D., Bouros, P.: Routing directions: keeping it fast and simple. In: Proceedings of the 21st ACM International Conference on Advances in Geographic Information Systems, pp. 164–173 (2013)
13. Westphal, M., Renz, J.: Evaluating and minimizing ambiguities in qualitative route instructions. In: Proceedings of the 19th ACM International Conference on Advances in Geographic Information Systems, pp. 171–180 (2011)
14. Lv, M., Chen, L., Chen, G.: Discovering personally semantic places from GPS trajectories. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, pp. 1552–1556 (2012)
15. Yan, Z., Chakraborty, D., Parent, C., Spaccapietra, S., Aberer, K.: Semitri: a framework for semantic annotation of heterogeneous trajectories. In: Proceedings of the 14th International Conference on Extending Database Technology, pp. 259–270 (2011)
16. Palma, A.T., Bogorny, V., Kuijpers, B., Alvares, L.O.: A clustering-based approach for discovering interesting places in trajectories. In: Proceedings of the ACM Symposium on Applied Computing, pp. 863–868 (2008)

17. Alvares, L.O., Bogorny, V., Kuijpers, B., de Macedo, J.A.F., Moelans, B., Vaisman, A.: A model for enriching trajectories with semantic geographical information. In: Proceedings of the 15th Annual ACM International Symposium on Advances in Geographic Information Systems, pp. 22:1–22:8 (2007)
18. Parent, C., Spaccapietra, S., Renso, C., Andrienko, G., Andrienko, N., Bogorny, V., Damiani, M.L., Gkoulalas-Divanis, A., Macedo, J., Pelekis, N., Theodoridis, Y., Yan, Z.: Semantic trajectories modeling and analysis. ACM Comput. Surv. **45**, 42:1–42:32 (2013)
19. Spaccapietra, S., Parent, C.: Adding meaning to your steps. In: Proceedings of the 30th International Conference on Conceptual Modeling, pp. 13–31 (2011)
20. Yan, Z., Chakraborty, D., Parent, C., Spaccapietra, S., Aberer, K.: Semantic trajectories: mobility data computation and annotation. ACM Trans. Intell. Syst. Technol. **4**, 49:1–49:38 (2013)
21. Yan, Z., Spremic, L., Chakraborty, D., Parent, C., Spaccapietra, S., Aberer, K.: Automatic construction and multi-level visualization of semantic trajectories. In: Proceedings of the 18th International Conference on Advances in Geographic Information Systems, pp. 524–525 (2010)
22. Feldman, D., Sugaya, A., Sung, C., Rus, D.: iDiary: from GPS signals to a text-searchable diary. In: Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems, pp. 6:1–6:12 (2013)