Inferring Semantics from Geometry - the Case of Street Networks

Padraig Corcoran
School of Computer Science & Informatics
Cardiff University
Cardiff, Wales, UK.
corcoranp@cardiff.ac.uk

Musfira Jilani
School of Computer Science & Informatics
University College Dublin
Dublin, Ireland.
musfira.jilani@ucdconnect.ie

Peter Mooney
Department of Computer Science
Maynooth University
Maynooth, Ireland.
peter.mooney@nuim.ie

Michela Bertolotto
School of Computer Science & Informatics
University College Dublin
Dublin, Ireland.
michela.bertolotto@ucd.ie

ABSTRACT

This paper proposes a method for automatically inferring semantic type information for a street network from its corresponding geometrical representation. Specifically, a street network is modelled as a *probabilistic graphical model* and semantic type information is inferred by performing learning and inference with respect to this model. Learning is performed using a *maximum-margin* approach while inference is performed using a *fusion moves* approach. The proposed model captures features relating to individual streets, such as linearity, as well as features relating to the relationships between streets such as the co-occurrence of semantic types. On a large street network containing 32,412 street segments, the proposed model achieves precision and recall values of 68% and 65% respectively. One application of this work is the automation of street network mapping.

Categories and Subject Descriptors

I.5 [**Pattern Recognition**]: Models—*Statistical*

General Terms

Theory

Keywords

Machine Learning, Geometry, Semantics

1. INTRODUCTION

Spatial information is a fundamental component of many applications such as assisted navigation [7] and information retrieval [31]. However, in order for such applications to be practical the spatial information in question must be of sufficient quality [9]. Spatial information quality lies along multiple dimensions; however the relevant dimensions ultimately depend on the nature of the data and the intended application. Longley et al. [23] suggested five general dimensions of attribute accuracy, positional accuracy, logical consistency, completeness and lineage. Temporal and semantic dimensions are also considered important [12].

Toward understanding the factors influencing spatial information quality, we consider the two predominant processes by which such information is created. The first process is through the use of traditional mapping practices and is employed by trained professionals working for proprietary data vendors and national mapping agencies. The second is through crowd-sourcing and is employed by projects such as OpenStreetMap (OSM) [25, 16]. The concept of crowd-sourcing spatial information was branded Volunteered Geographic Information (VGI) by Goodchild [10] and this has become an accepted term in the Geographical Information Science community. Depending on which of these processes is employed, a number of different factors may influence the corresponding spatial information quality. For example, contributors to crowd-sourcing projects generally do not have any formal training with respect to good mapping practices and this can negatively affect logical consistency. On the other hand, individuals working for proprietary data vendors and national mapping agencies generally will not have in-depth geographical knowledge of the areas they map and this can negatively affect semantic accuracy. For the purpose of this paper, we define semantics to be the type information associated to geographical features. However, irrespective of the process employed, one factor which influences quality with respect to multiple dimensions is that creating spatial information is labour demanding. In fact, empirical analysis of OSM has demonstrated that there exists a positive correlation between the number of contributors in

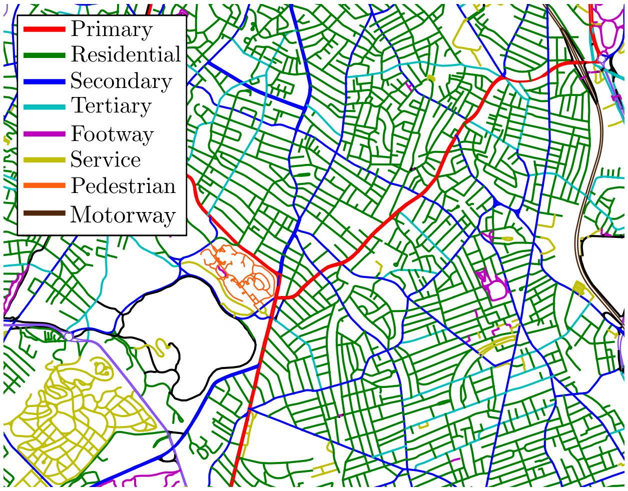

Figure 1: A subset of the OSM street network representation for the city of Boston Massachusetts. Each street segment is represented by a colour indicating its semantic type and this mapping is given in the contained legend.

an area and the corresponding data quality [13, 28]. Therefore, if the above mapping processes could be automated to a greater degree this would reduce labour requirements and in turn positively impact information quality.

Toward this goal, a number of approaches to automating the process of creating spatial information have been considered. One class of approaches is based on the hypothesis that much of a map's semantic type information is implicitly represented in the corresponding geometry of the map and therefore may be automatically inferred [36]. To date such works have mainly focused on inferring semantic information relating to buildings [32, 19]. Determining whether such an approach can be applied in the context of street networks has yet to be considered and is the focus of this paper. Toward motivating this work consider Figure 1 which displays a subset of the OSM street network representation for the city of Boston Massachusetts where the colour of each street segment indicates its semantic type. In this context semantic type equals the corresponding OSM *highway* tag value where a description of the different tag values can be found on the OSM wiki [1]. An examination of Figure 1 reveals that primary streets generally exhibit characteristics of being quite long, linear and having dual-lanes. On the other hand, secondary streets also exhibit characteristics of generally being quite long and linear but generally do not have dual-lanes. Residential streets generally exhibit the characteristic of being shorter than primary or secondary streets and exhibit a grid like structure. Pedestrian streets generally exhibit characteristics of being quite short, non-linear and unstructured.

The above discussion implies that, through the application of a machine learning paradigm, it may be possible to learn a model relating the geometry and semantic type information of a street network. Such a model can then in turn be used to infer semantic type information from geometry. Toward this goal we propose to model a street network as a probabilistic graphical model where semantic type information may be inferred by performing learning and inference with respect to this model. The proposed model captures the geometrical characteristics of individual streets (e.g. length) through node potentials and relations between streets (e.g. relative orientation) through pairwise potentials. It also allows efficient inference [22] and is trained using a maximum-margin approach [18].

The layout of this paper is as follows. In section 2 we review related works on the topic of inferring semantic information of geographical objects using a machine learning paradigm. Section 3 describes the proposed probabilistic graphical model along with corresponding learning and inference methods employed. Section 4 presents a set of experiments which evaluate the proposed model. Finally in section 5 we draw conclusions from the work presented and discuss possible future research directions.

2. RELATED WORKS

Lüscher et al. [24] proposed a method to infer semantic type information corresponding to buildings. The authors use a Bayesian network or directed graphical model where the structure of the model is taken from a previously defined ontology. Learning is performed in a nonparametric manner using kernel density estimation while inference is performed using belief propagation. This model uses a number of geometrical and topological features which are defined by the ontology. Huang et al. [15] proposed a Markov random field model for inferring semantic type information corresponding to buildings. Inference is performed using Gibbs sampling. No learning is performed and instead the model parameters are manually set. The proposed model uses a number of features which model the shape of building footprints. Using similar shape features, Fan et al. [8] proposed a rule based approach to infer semantic type information corresponding to buildings. Sester [32] proposed a decision tree model for discriminating between houses, streets and land parcels. The model was trained using the ID3 algorithm and uses geometrical and topological features. Keyes et al. [19, 20] performed supervised and unsupervised classification of buildings using the shape features of Fourier descriptors, moment invariants and scalar descriptors. Walter and Luo [34] proposed a neural network model for discriminating between different types of geographical objects including streets and buildings. The model uses a number of geometrical features relating to the size and shape of objects. Results achieved are very positive but correspond to a very small dataset. Henn et al. [14] used a Support Vector Machine model to infer semantic type information corresponding to buildings in three dimensional city models. A large number of geometrical (e.g. area and width) and semantic (e.g. distance to nearest hospital) were used by the model. Werder et al. [35] proposed a method for classifying buildings and city blocks in an unsupervised manner using a mixture of Gaussians model and performed inference using the EM algorithm. The model uses two sets of features which characterize individual buildings and city blocks.

3. METHODOLOGY

This section is structured as follows. Section 3.1 presents the street network representation used within this work.

[1]http://wiki.openstreetmap.org/wiki/Key:highway

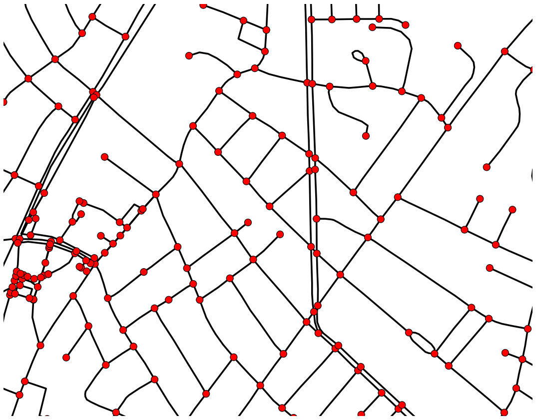

Figure 2: A example of the street network representation $G^s = (V^s, E^s)$ is illustrated. Elements of the sets V^s and E^s are represented by red circles and black lines respectively.

Section 3.2 describes how this representation is in turn modelled using probabilistic graphical model. Section 3.3 describes the features captured by this model. Finally sections 3.4 and 3.5 describe the methods employed for performing learning and inference with respect to this model respectively.

3.1 Street Network Representation

We represent a given street network using a graph $G^s = (V^s, E^s)$ where the set of vertices V^s correspond to street intersections and deadends, while the set of edges E^s correspond to street segments connecting these vertices. An example of this representation is illustrated in Figure 2. This is a commonly used street network representation and is known as a *primary representation* [29, 6]. A UTM coordinate system, where distances measured are in meters, was used to represent the spatial locations of all vertices and edges in G^s.

3.2 Model formulation

We model a given street network as a Markov random field with log-linear node and pairwise edge potentials [27]. Let $x = (x_1, \ldots, x_N)$ be a set of random variables corresponding to the set of N street segments described in the previous section and let $y = (y_1, \ldots, y_N)$ be a labelling of these segments with respect to semantic type. Each element y_i is a vector of length K, $y_i = (y_i^1, \ldots, y_i^K)$, where K is the number of possible street labels and each $y_i^k \in \{0, 1\}$ indicates if segment i is assigned label k. For a given i, only a single y_i^k can be assigned a value of 1.

For a given x, the prediction $\hat{y}$ as defined by Equation 1 is the arg max of the discriminant function $f_{\mathbf{w}} : \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}$ which is parametrized by a vector of weights w.

$$\hat{y} = \arg\max_{y \in \mathcal{Y}} f_{\mathbf{w}}(x, y) \tag{1}$$

The discriminant function $f_w(x, y)$ captures the dependencies between street segments and labels as follows. Let $G^c = (V^c, E^c)$ be a graph where a bijection exists between the set of vertices V^c and the set x. An edge is constructed

Table 1: Vertex and edge features

Vertex features
Degree
Length
Linearity
Parallelism

Edge features
Relative labels
Turning angle

between two vertices in G^c if the corresponding street segments are adjacent (share an end point). A Markov property with respect to G^c is assumed which states that each vertex in G^c is independent of all other vertices given its neighbouring vertices. Given this assumption, we factor $f_w(x, y)$ as defined in Equation 2 where the y_v denotes the element of y corresponding to v and $\sim$ denotes adjacency between vertices.

$$f_{\mathbf{w}}(x, y) = \sum_{v \in V^c} \sum_{i=1}^{K} y_v^i w_n^i \phi_n(v) + \sum_{(u,v \in V^c; u \sim v)} \sum_{i=1}^{K} \sum_{j=1}^{K} y_v^i y_u^j w_t^{ij} \phi_t(u, v) \tag{2}$$

The terms $\phi_n(v)$ and $\phi_t(u, v)$ in Equation 2 correspond to vertex and edge feature vectors respectively. The vertex feature vector $\phi_n(v)$ describes the street segment in question. Elements of this vector include the geometrical shape of the segment. The edge feature vector $\phi_t(u, v)$ describes the relationship between the street segments in question. Elements of this vector include the relative orientation of the segments. The terms w_n^i and w_t^{ij} correspond to weight vectors. The term w_n^i models the dependencies between label i and the vertex feature vector. The term w_t^{ij} models the dependencies between labels i, j and the edge feature vector. The vector $\mathbf{w}$ in $f_{\mathbf{w}}(x, y)$ of Equations 1 and 2 corresponds to an appropriate stacking of the vectors w_n^i and w_t^{ij}.

3.3 Features

The properties of street segments and the relations between these segments are captured by our model with the aid of computed features. As discussed in section 3.2 our model contains both a vertex feature vector $\phi_n(v)$ and an edge feature vector $\phi_t(u, v)$. The elements of these vectors are listed in Table 1 and described in the following subsections.

3.3.1 Degree & Length

The degree of a street segment is the number of other segments it is directly connected to. This features captures an aspect of the local complexity of the network. The length feature corresponds to the length of the street segment in question measured in meters.

3.3.2 Linearity

This feature captures the degree to which the street segment in question belongs to a linear street and is computed as follows. Let S be a sequence of street segments which initially only contains the segment for which we wish to com-

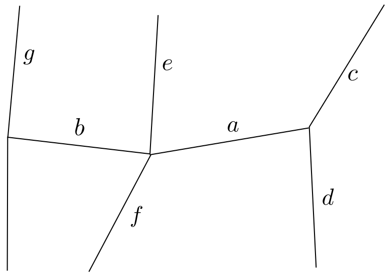

Figure 3: A simple street network is illustrated where each street segment is represented by a line segment and assigned a unique label. In the case of computing the linear feature for the street segment a, the sequence S initially contains a single element of a; that is $S = \{a\}$. In the first iteration of the procedure for computing the linear feature the street segment b is added to the start of S; that is $S = \{b, a\}$. In the second iteration the street segment c is added to the end of S; that is $S = \{b, a, c\}$.

pute its linearity. S is iteratively expanded such that at each iteration that segment adjacent to the first or last segments in S which results in the most linear sequence of segments is added to start or end of S respectively. Using a simple street network, two iterations of this procedure are illustrated in Figure 3. The search is terminated when the length of the sequence of segments in S exceeds a threshold t. Setting the threshold t to a particular value has the effect of encouraging the extraction of linear streets of a length greater than or equal to that value. The linearity of a sequence of segments S, denoted $L(S)$, is computing as follows. First the distance d in terms of shape between S and a line segment is computed using the metric of [2]. Next the linearity of S is computed by scaling d to the range $[1, 0)$ using the function in Equation 3. In this work we computed linear features of multiple scales for each street segment using t values of 500, 1000, 2000, and 4000.

$$L(S) = \frac{1}{1+d} \tag{3}$$

To illustrate this feature consider the sample street network represented in Figure 4 which is part of a much larger street network corresponding to the cities of Boston and Cambridge in Massachusetts. The linear feature of each street segment in this network, where the value of t is 2000, is represented in Figure 5. From this figure it is evident that long linear streets exhibit the feature in question.

3.3.3 Parallelism

This feature captures the degree to which there exists one or more streets which are parallel to the street segment in question. We now describe how this feature is computed for a single street segment a. Let N be the set of street segments which lie within the spatial neighbourhood of, but are not adjacent to, the segment a. This set is illustrated in Figure 6. In our implementation a street segment is determined to lie within the neighbourhood of another segment if the distance between any pair of points, where a single point belongs to each segment, is less than 350 meters. Neighbourhood queries are performed with the aid of a KD-tree

Figure 4: Part of a much larger street network corresponding to the cities of Boston and Cambridge in Massachusetts is displayed.

Figure 5: The linearity feature of each street segment in Figure 4 is represented by a grayscale map where black corresponds to a value of one while colours approaching white correspond to a value of zero.

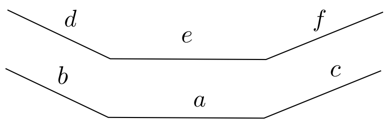

Figure 6: A simple street network is illustrated where each street segment is represented by a line segment and assigned a unique label. In the case of computing the parallelism feature for the street segment a, the set N is equal to $\{d, e, f\}$.

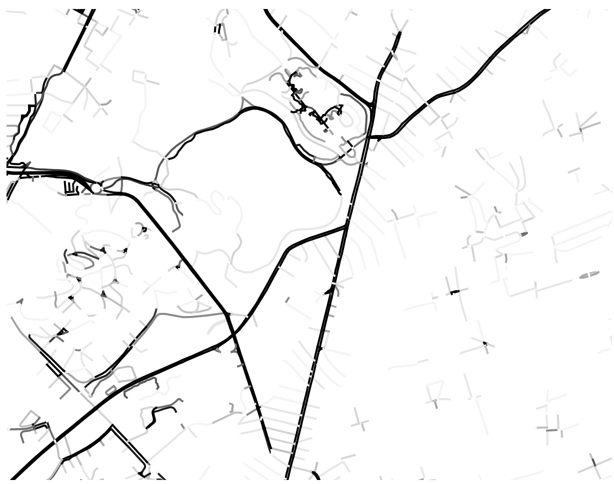

Figure 7: The parallelism feature of each street segment in Figure 4 is represented by a grayscale map where black corresponds to a value of one while colours approaching white correspond to a value of zero.

data structure. Next, let P be the set of points obtained by sampling a point every 10 meters along the segment a. The shortest distance from each point in P to the street segments contained in N is computed. Finally the percentage of these distances less than 35 meters is used as the parallelism feature.

To illustrate this feature again consider the sample street network representation in Figure 4. The parallelism feature of each street segment in this network is represented in Figure 7. From this figure it is evident that multi-lane streets exhibit the feature in question.

3.3.4 Relative labels

In a street network, some pairs of adjacent street segment labels are more likely than others. For example, it is unlikely that the pair of street segment labels residential and motorway will be adjacent. On the other hand, it is very likely that the pair of street segment labels residential and residential will be adjacent. In order to capture this characteristic we use a feature which has a constant value of one. The corresponding weights w_t^{ij} in Equation 2 when learnt will represent a bias towards pairs of adjacent street segment labels which are more likely to occur.

3.3.5 Turning angle

The turning angle between adjacent street segments corresponds to the relative difference in orientation of the line segments formed by joining the first and last points of each segment. This concept is illustrated in Figure 8. The use of this feature is motivated by the fact that different pairs of adjacent street segment labels are likely to have different turning angles. For example, it is unlikely that a pair of adjacent street segments both with the label motorway will have a corresponding large turning angle. On the other hand, it is quite likely that a pair of adjacent street segments both with the label residential could have a corresponding turning angle which is large or small.

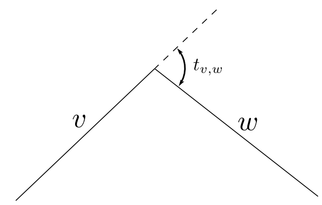

Figure 8: Two line segments v and w, and the turning angle $t_{v,w}$ between these segments is illustrated.

3.4 Learning

Learning corresponds to determining the weight vector $\mathbf{w}$ from training examples $(x_1, y_1), \ldots, (x_n, y_n)$. To simplify notation we rewrite Equation 2 as Equation 4 using an appropriate stacking of the vectors w_n^i and w_t^{ij} into $\mathbf{w}$ and the vectors $\phi_n(v)$ and $\phi_t(u, v)$ into $\Psi(x, y)$.

$$f_{\mathbf{w}}(x, y) = \mathbf{w}^T \Psi(x, y) \tag{4}$$

The weight vector $\mathbf{w}$ is learned using a maximum-margin approach through the minimization of the constrained quadratic optimization problem of Equation 5 [33, 17].

$$\begin{aligned} & \min_{\mathbf{w}, \xi \geq 0} \frac{1}{2} w^T w + C\xi \\ & s.t. \;\; \forall (\bar{y}_1, \ldots, \bar{y}_n) \in \mathcal{Y}^n : \\ & \frac{1}{n} w^T \sum_{i=1}^{n} [\Psi(x_i, y_i) - \Psi(x_i, \bar{y}_i)] \geq \Delta(y_i, \bar{y}_i) - \xi \end{aligned} \tag{5}$$

The term ξ is a slack variable while $w^T w$ represents the size of the margin dividing the training examples. The term C balances these two terms in the objective. The term $\Delta(y_i, \bar{y}_i)$ is a function which measures the loss associated with a labelling $\bar{y}_i$ if the true labelling is y_i. In this work the Hamming loss as defined in Equation 6 is used. Intuitively, the above optimization problem models the fact that we wish to minimize the number of incorrect classifications while maximizing the margin.

$$\Delta(y_i, \bar{y}_i) = \sum_{i=1}^{N} \sum_{k=1}^{K} \left| y_i^k - \bar{y}_i^k \right| \tag{6}$$

The number of constraints in Equation 5 is exponential in the dimensionality of y. Despite this fact, Joachims et al. [17] proved that an ϵ-accurate solution may be found in polynomial time via a cutting plane algorithm. This algorithm constructs a working subset of the constraints that is built incrementally by adding the most violated constraint for each training example [18]. The search for the most violated constraint with respect to the ith training example can be formulated as the optimization problem of Equation 7. This represents a quadratic program (QP) which can be solved in polynomial time [4]. Specifically we use the QP solver of [11].

$$\hat{y}_i = \underset{y \in \mathcal{Y}}{\arg\max} \left(\mathbf{w}^T \Psi(x_i, y) + \Delta(y_i, y) \right) \tag{7}$$

3.5 Inference

The task of inference consists of labelling each segment in a given street network with the most appropriate label. That is, compute the solution to Equation 1. However this problem is NP-hard and therefore an optimal solution cannot be found efficiently. In this work we use an approximation to the solution using an iterative technique known as α-expansion [5]. At each iteration, a single label value α is considered, and for every street segment, its current label value is either retained, or changed to α. The name α-expansion derives from the fact that the set of street segments with label value α expands at each iteration. The process is iterated until no choice of α value causes any change. Each expansion iteration is guaranteed to lower the overall objective function; however the final result is not guaranteed to be the optimal solution.

Determining the optimal α-expansion at each iteration corresponds to solving the following fusion moves problem [22]. Let $x^1 = (x^1_1, \ldots, x^1_N)$ be the current labelling of the street segments and $x^2 = (x^2_1, \ldots, x^2_N)$ a labelling where each street segment is assigned the label α. Let $s = (s_1, \ldots, s_N)$ be a set of binary variables where a bijection exists between y and the set of street segments. Determine the optimal α-expansion reduces to determining the optimal fusion move $x^c(s)$, as defined by Equation 8, which is a function of s.

$$x^c(s) = (1 - s^T)x^1 + (s^T)x^2 \tag{8}$$

This is achieved by solving a particular linear programming (LP) relaxation where the binary values are replaced by values in the range $[0, 1]$. This relaxation is then solved using quadratic pseudo-boolean optimization (QPBO) [21, 30, 22, 26]. Specifically, we use the QPBOI implementation of Rother et al. [30].

4. EXPERIMENTS

The structure of this section is as follows. Section 4.1 describes the data used for evaluation. Section 4.2 describes the performance metrics considered. Finally in section 4.3 we present results.

4.1 Data

Two street network representations obtained from OpenStreetMap(OSM) were used in our experiments. These correspond to non-intersecting sections of the street network representation for the cities of Boston and Cambridge in Massachusetts. The street network representation used for evaluation is visualized in Figure 9. This network has a corresponding primal representation containing 22,458 vertices and 32,412 edges. As such, it represents an extremely large dataset. The other street network representation was used for model learning, and contains 16,368 vertices and 25,215 edges in its primal representation. This dataset corresponds to a geographical area located north of that represented in Figure 9.

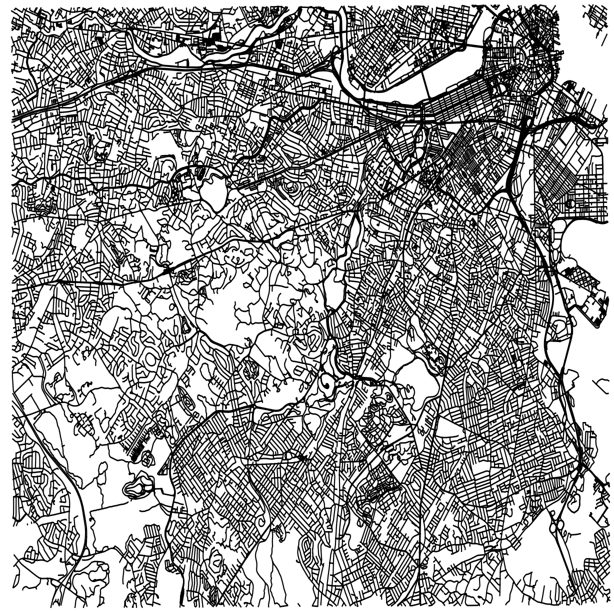

Figure 9: The OpenStreetMap street network representation for the cities of Boston and Cambridge is displayed.

The OSM street network for the whole of the US has received high quality bulk data imports in the form of Tiger data from the US Census Bureau [37]. As such, the OSM street network for the chosen study area can be considered to be of high geometric and semantic quality. Although the proposed model for determining semantic type information is a function of only geometrical features, a high semantic quality is necessary for learning and to act as ground truth data within our evaluation. In OSM there exists no restriction on the semantic type a contributor can assign to a street. However, toward maintaining consistency, OSM defines a large set of semantic types which contributors are encouraged to use. The OSM street network used within this evaluation contains streets of 27 different semantic types. However most of these types only represent a small percentage of the overall street network. For example, the OSM street network used for evaluation contains only 4 street segments of the semantic type 'platform'. Therefore we only considered the following eight semantic types which are significantly represented in the corresponding street network: 'Primary', 'Tertiary', 'Service', 'Residential', 'Secondary', 'Motorway', 'Trunk' and 'Footway'. A description of the meaning of each of these types can be found on the OSM wiki [2]. This set of types constitutes 30,650 of the 32,412 (95 %) street segments contained in the network used for evaluation. The remaining 1761 (5 %) street segments correspond to 18 different street types which include *pedestrian* and *steps*.

[2]http://wiki.openstreetmap.org/wiki/Key:highway

	Primary	Tertiary	Service	Residential	Secondary	Motorway	Trunk	Footway
Primary	510	0	0	33	541	0	49	64
Tertiary	23	512	0	1012	886	3	9	37
Service	39	1	234	2498	71	0	12	1241
Residential	84	17	47	12288	906	0	32	499
Secondary	125	24	10	855	3581	0	96	225
Motorway	0	0	0	0	0	77	0	3
Trunk	19	0	0	30	216	0	397	60
Footway	56	5	27	647	114	3	31	2401

Table 2: Confusion Matrix for the proposed model.

(a) (b)

Figure 10: Street segment classes determined by the proposed model and correspond ground truth classes for a subset of the street network used for evaluation are displayed in (a) and (b) respectively. See figure 1 for the corresponding legend.

4.2 Performance metrics

In order to quantify the performance of the proposed model we employed both confusion matrices and the statistics of precision and recall. These statistics are defined as follows. Let $pred_i(k)$ denote that the street indexed by i was determined to be of class k and $gt_i(k)$ denote that the street indexed by i has ground truth label k. The precision and recall corresponding to class k are defined in Equations 9 and 10 respectively.

$$precision(k) = \frac{|\{i : pred_i(k) \wedge gt_i(k)\}|}{|\{i : pred_i(k)\}|} \tag{9}$$

$$recall(k) = \frac{|\{i : pred_i(k) \wedge gt_i(k)\}|}{|\{i : gt_i(k)\}|} \tag{10}$$

In order to generate summary statistics of all the individual precision and recall values, we employed weighted precision and recall. These statistics are defined in Equation 11 and 12 respectively and perform a weighing by class support.

$$weighted\_percision = \frac{\sum_k^K percision(k) \times |\{i : gt_i(k)\}|}{\sum_k^K |\{i : gt_i(k)\}|} \tag{11}$$

$$weighted\_recall = \frac{\sum_k^K recall(k) \times |\{i : gt_i(k)\}|}{\sum_k^K |\{i : gt_i(k)\}|} \tag{12}$$

4.3 Results

Tables 2 and 3 display the confusion matrix and precision/recall values corresponding to the proposed model respectively. Weighted precision and recall values of 68% and 65% respectively are achieved. This result is very positive and supports the hypothesis set forth in this paper that much of a street networks semantic information may be automatically inferred from the geometrical representation of the network. A closer examination of the result reveals some important attributes regarding the proposed model. It is evident that the proposed model performs extremely well with respect to the semantic types of Residential and Motorway. In both cases the high precision and recall values are achieved. On the other hand, performance was poorest with respect to the semantic types of Primary, Tertiary and Service. Examining the confusion matrix reveals that this can be attributed to the fact that the proposed model cannot distinguish effectively between these classes.

Figures 10(a) and 10(b) display the class determined by the proposed model and corresponding ground truth respectively for a subset of the street network used for evaluation. Many of the properties exhibited by the confusion matrix of Table 2 are reflected in these results. For example, the model performed very well with respect to Residential streets. Also a significant percentage of the Tertiary streets are classified as Secondary streets.

In order to determine the importance of the edge features (see Table 1), which consider context information such as relative label values, we performed learning and inference using only vertex features. In this case the proposed model reduces to a multi-class Support Vector Machine (SVM) [17]. The precision and recall values corresponding to this reduced model are listed in Table 4. We observe from these results that the reduced model performed significantly poorer than the original model. This in turn indicates the importance of the edge features.

	Precision	Recall	Support
Primary	0.60	0.43	1197
Tertiary	0.92	0.21	2482
Service	0.74	0.06	4096
Residential	0.71	0.89	13873
Secondary	0.57	0.73	4916
Motorway	0.93	0.99	80
Trunk	0.63	0.55	722
Footway	0.53	0.73	3284
Weighted	0.68	0.65	30650

Table 3: Precision and recall values for the proposed model. Support indicates the number of street segments of the class in question.

	Precision	Recall	Support
Primary	0.86	0.22	1197
Tertiary	0.32	0.01	2482
Service	0.36	0.15	4096
Residential	0.61	0.90	13873
Secondary	0.46	0.38	4916
Motorway	0.75	0.45	80
Trunk	0.46	0.06	722
Footway	0.48	0.56	3284
Weighted	0.52	0.56	30650

Table 4: Precision and recall values for the reduced model. Support indicates the number of street segments of the class in question.

5. CONCLUSIONS

We live in an era where the need for accurate and current spatial data has never been greater. We argue that one strategy toward fulfilling this need is to automate the mapping process as much as possible. The work presented in this paper represents a step toward that goal. Specifically a method for inferring semantic type information for a street network from the corresponding geometrical representation of the network is proposed. This is achieved by modelling the network as a probabilistic graphical model and performing learning and inference with respect to this model. On average results obtained were very positive with the proposed model achieving weighed precision and recall values of 68% and 65% respectively on a large evaluation street network.

Despite these positive results there exists much potential to improve and build upon the research presented here. Toward improving the performance of the proposed model, implementing the following two strategies could prove worthwhile. Firstly extracting a greater array of features might allow a greater ability to discriminate between the different semantic types. One avenue to implementing this strategy would be to draw from feature extraction techniques in the domain of computer vision. For example, shape context [3] has been shown to accurately model the shapes of objects and could prove to be an accurate model of the shape of street segments and their neighbourhoods. A second strategy toward improving performance would be to reduce the number of semantic types the model attempts to discriminate between by merging those types which the model struggles to discriminate between. This strategy has previously been employed in related works such as that of [35].

The model proposed in this paper could also provide a platform for related research enterprises. For example, there exists a large number of works on the topic of map construction which attempt to infer the geometry of a street network from GPS trajectories [1]. We hypothesise that it may potentially be possible to integrate work in this domain with that presented in this paper toward the development of a methodology capable of inferring both the geometry and semantic of a street network from GPS trajectories.

6. ACKNOWLEDGEMENTS

This work was supported in part by a European Marie Curie International Outgoing Fellowship (Project number 327710) and an Irish Research Council Government of Ireland Postgraduate Scholarship.

7. REFERENCES

[1] M. Ahmed, S. Karagiorgou, D. Pfoser, and C. Wenk. A comparison and evaluation of map construction algorithms using vehicle tracking data. *GeoInformatica*, pages 1–32, 2014.

[2] E. M. Arkin, L. P. Chew, D. P. Huttenlocher, K. Kedem, and J. S. Mitchell. An efficiently computable metric for comparing polygonal shapes. In *Proceedings of the first annual ACM-SIAM symposium on Discrete algorithms*, pages 129–137. Society for Industrial and Applied Mathematics, 1990.

[3] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002.

[4] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239, 2001.

[6] P. Corcoran, P. Mooney, and M. Bertolotto. Analysing the growth of openstreetmap networks. *Spatial Statistics*, 3:21–32, 2013.

[7] P. Corcoran, P. Mooney, and M. Bertolotto. Interactive cartographic route descriptions. *GeoInformatica*, 18(1):1–26, 2014.

[8] H. Fan, A. Zipf, and Q. Fu. *Connecting a Digital Europe Through Location and Place*, chapter Estimation of Building Types on OpenStreetMap Based on Urban Morphology Analysis, pages 19–35. Lecture Notes in Geoinformation and Cartography. Springer, 2014.

[9] S. Funke, R. Schirrmeister, and S. Storandt. Automatic Extrapolation of Missing Road Network Data in OpenStreetMap. In *ICML Workshop Mining Urban Data 2*, 2015.

[10] M. Goodchild. Citizens as sensors: the world of volunteered geography. *GeoJournal*, 69(4):211–221, 2007.

[11] M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. `http://cvxr.com/cvx`, 2014.

[12] S. C. Guptill, J. L. Morrison, I. C. Association, et al. *Elements of spatial data quality*. Elsevier, 1995.

[13] M. Haklay, S. Basiouka, V. Antoniou, and A. Ather. How many volunteers does it take to map an area well? the validity of linus; law to volunteered geographic information. *The Cartographic Journal*, 47(4):315–322, 2010.

[14] A. Henn, C. Römer, G. Gröger, and L. Plümer. Automatic classification of building types in 3d city models. *GeoInformatica*, 16(2):281–306, 2012.

[15] H. Huang, B. Kieler, and M. Sester. Urban building usage labeling by geometric and context analyses of the footprint data. In *Proceeding of 26th international cartographic conference (ICC)*, Dresden, Germany, 2013.

[16] M. Jilani, P. Corcoran, and M. Bertolotto. Automated highway tag assessment of openstreetmap road networks. In *Proceedings of the 22Nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '14, pages 449–452, New York, NY, USA, 2014. ACM.

[17] T. Joachims, T. Finley, and C. Yu. Cutting-plane training of structural SVMs. *Machine Learning*, 77(1):27–59, 2009.

[18] T. Joachims, T. Hofmann, Y. Yue, and C.-N. Yu. Predicting structured objects with support vector machines. *Communications of the ACM*, 52(11):97–104, 2009.

[19] L. Keyes and A. Winstanley. Using moment invariants for classifying shapes on large-scale maps. *Computers, Environment and Urban Systems*, 25(1):119–130, 2001.

[20] L. Keyes, A. Winstanley, and P. Healy. Comparing learning strategies for topographic object classification. In *IEEE international Geoscience and remote sensing symposium*, volume 6, pages 3468–3470. IEEE, 2003.

[21] V. Kolmogorov and C. Rother. Minimizing nonsubmodular functions with graph cuts-a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(7):1274–1279, 2007.

[22] V. Lempitsky, C. Rother, S. Roth, and A. Blake. Fusion moves for markov random field optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1392–1405, 2010.

[23] P. Longley, M. Goodchild, D. Maguire, and D. Rhind. *Geographic information systems and science*. John Wiley & Sons, 2005.

[24] P. Lüscher, R. Weibel, and D. Burghardt. Integrating ontological modelling and bayesian inference for pattern classification in topographic vector data. *Computers, Environment and Urban Systems*, 33(5):363–374, 2009.

[25] P. Mooney and P. Corcoran. The annotation process in openstreetmap. *Transactions in GIS*, 16(4):561–579, 2012.

[26] A. C. Müller and S. Behnke. Pystruct: learning

structured prediction in python. *The Journal of Machine Learning Research*, 15(1):2055–2060, 2014.

[27] K. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[28] P. Neis, D. Zielstra, and A. Zipf. The street network evolution of crowdsourced maps: Openstreetmap in germany 2007–2011. *Future Internet*, 4(1):1–21, 2011.

[29] S. Porta, P. Crucitti, and V. Latora. The network analysis of urban streets: a dual approach. *Physica A: Statistical Mechanics and its Applications*, 369(2):853–866, 2006.

[30] C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer. Optimizing binary mrfs via extended roof duality. In *IEEE conference on Computer vision and pattern recognition*, pages 1–8, 2007.

[31] H. Samet, J. Sankaranarayanan, M. D. Lieberman, M. D. Adelfio, B. C. Fruin, J. M. Lotkowski, D. Panozzo, J. Sperling, and B. E. Teitler. Reading News with Maps by Exploiting Spatial Synonyms. *Communications of the ACM*, 57(10):64–77, 2014.

[32] M. Sester. Knowledge acquisition for the automatic interpretation of spatial data. *International Journal of Geographical Information Science*, 14(1):1–24, 2000.

[33] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of the twenty-first international conference on Machine learning*, page 104. ACM, 2004.

[34] V. Walter and F. Luo. Automatic interpretation of digital maps. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(4):519–528, 2011.

[35] S. Werder, B. Kieler, and M. Sester. Semi-automatic interpretation of buildings and settlement areas in user-generated spatial data. In *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*, pages 330–339. ACM, 2010.

[36] X. Zhang, T. Ai, J. Stoter, M.-J. Kraak, and M. Molenaar. Building pattern recognition in topographic data: examples on collinear and curvilinear alignments. *GeoInformatica*, 17(1):1–33, 2013.

[37] D. Zielstra, H. H. Hochmair, and P. Neis. Assessing the effect of data imports on the completeness of openstreetmap–a united states case study. *Transactions in GIS*, 17(3):315–334, 2013.