Comparing Vessel Trajectories using Geographical Domain Knowledge and Alignments

Gerben K.D. de Vries G.K.D.deVries@uva.nl
Informatics Institute, University of Amsterdam, Sciencepark 904, 1098 XH, Amsterdam, the Netherlands

Willem Robert van Hage wrvhage@few.vu.nl
Computer Science, VU University Amsterdam, de Boelelaan 1081a, 1081 HV, Amsterdam, the Netherlands

Maarten van Someren M.W.vanSomeren@uva.nl
Informatics Institute, University of Amsterdam, Sciencepark 904, 1098 XH, Amsterdam, the Netherlands

1. Introduction

In this paper we present an alignment based similarity measure that combines low-level vessel trajectories with geographical domain knowledge, such as the name and type of the regions that vessels pass through and stop. We use this similarity measure in a clustering experiment to discover interesting behavior and in a classification task to predict the type of the vessel for a trajectory. The combination of information gives the best average classification accuracy. For both clustering and classification we use kernel based algorithms.

2. Trajectories & Geographical Domain Knowledge

We define a trajectory as $T = \langle x_1, y_1 \rangle, \ldots, \langle x_n, y_n \rangle$, ignoring the temporal dimension. The number of vectors in T is denoted as: $|T|$. In the *stop* and *move* model of (Spaccapietra et al., 2008), the trajectories in our experiment are moves. They are delimited by the vessel entering the area of observation or starting, and the vessel leaving the area of observation or stopping.

The geographical domain knowledge comes as two simple ontologies. One, **A**&**C**, contains the definitions of different anchorages, clear ways, and other areas at sea. The other ontology, **H**, defines different types of harbors, such as liquid bulk and general cargo. For both ontologies, we created a SWI-Prolog webservice (van Hage et al., 2010) to enrich vessel trajectories with geographical features. The first service returns a set of specific type, label pairs corresponding to the regions in **A**&**C** that intersect with a given point. We create a sequence of sets of geo-labels $T^L = L_1, \ldots, L_{|T|}$ for a trajectory T with this service. For the start and end of a trajectory we define objects that contain information whether the vessel is stopped and if so at what harbor or region. We discover this harbor using the second webservice, which matches a point to the nearest harbor in **H** that is within range and returns the label and specific type of this harbor. If there is no harbor close, we use the first webservice.

3. Trajectory Similarity

For the sequences T and T^L we compute similarity using an edit distance alignment, which we discovered in previous work (de Vries & van Someren, 2010) to perform the best on a vessel trajectory clustering task. To compute an edit distance, we need a substitution function and a gap penalty. The substitution function for trajectories T is defined as: $\mathbf{sub}_{\text{traj}}(\langle x_i, y_i \rangle, \langle x_j, y_j \rangle) = -\|\langle x_i - x_j, y_i - y_j \rangle\|$, i.e. the negative of the Euclidean distance. We take the value for the gap penalty g from the mentioned previous work. For T^L, the substitution function $\mathbf{sub}_{\text{lab}}(L_i, L_j)$ expresses how many labels the sets of labels L_i and L_j have in common. We set g as the minimally possible $\mathbf{sub}_{\text{lab}}$ score.

The similarity $Sim(S, T)$ between two sequences S and T is the score of the alignment that has the maximum edit distance score for all possible alignments between these sequences, divided by $|S| + |T|$ to give the average score per element. In the experiments we use kernel based algorithms. For all sequences T_i and T_j in a set of sequences $\mathcal{T}$, we compute a kernel K as: $K(i, j) = Sim(T_i, T_j)$, then we normalize K and turn it into a kernel by $K = 1 - \frac{K}{\min(K)}$. For trajectories T we get a kernel K_{traj} and for sequences of sets of geo-labels T^L we get a kernel K_{lab}. The similarity between two start/end objects can immediately be put into kernel form and is determined by whether the vessel is stopped or not and how much la-

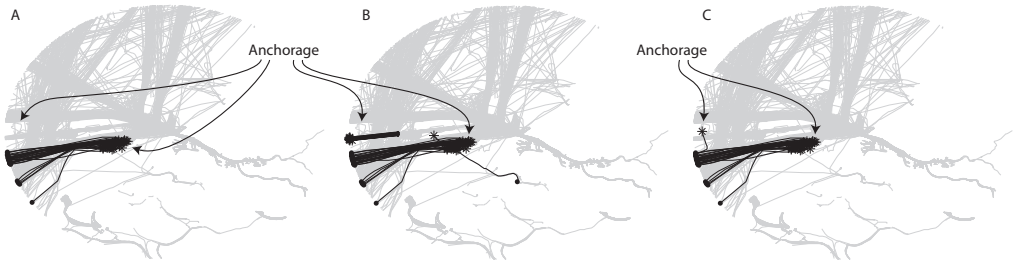

Figure 1. Example of a cluster of trajectories showing anchoring behavior. The example cluster is shown in black against the entire dataset in gray. The start of trajectory is indicated by a dot, the end by an asterix.

bels there are in common. This gives us a kernel K_{start} for the start objects and a kernel K_{end} for the end objects. $K_{\text{all}} = w_1 K_{\text{traj}} + w_2 K_{\text{lab}} + w_3 K_{\text{start}} + w_4 K_{\text{end}}$ combines the four kernels above. Clearly, this kernel is symmetric, but it is not guaranteed to be positive semi-definite.

4. Experiments

Our experimental dataset consists of 1917 vessel trajectories in a 50km radius area around the Port of Rotterdam, collected using the Automatic Identification System (AIS). The trajectories are compressed with the algorithm in (Gudmundsson et al., 2009), reducing the data by 95%, thus reducing computation time drastically. This compression improves performance on a vessel trajectory clustering task (de Vries & van Someren, 2010) using the same alignment.

For the clustering experiment we used weighted kernel k-means (Dhillon et al., 2007), with $k = 40$. We created kernels for 3 different weight settings of K_{all}: equal combination of domain knowledge and raw trajectories, K_{comb}, only raw trajectory information, K_{raw}, and only domain knowledge, K_{dom}. This results in a number of interesting clusters. In Figure 1A we see a cluster from clustering with K_{comb} that shows trajectories that enter the area from the west and anchor in one specific anchoring area. In B and C we plotted the most similar cluster from clustering with K_{raw} and K_{dom}, respectively. In Figure 1B there are also trajectories included that do not show the anchoring behavior, because we only consider raw trajectory information. We see the opposite in Figure 1C, where we have only anchoring behavior, but in different anchoring areas.

We also did a classification experiment, predicting the vessel's type. In total there are 18 types, available from AIS. For classification we used a support vector machine (SVM), with the same kernels as for clustering, in a 10-fold cross validation set-up. The classification accuracy for K_{all} was 75.4%, for K_{raw} 72.2%, and for K_{dom} 66.1%. All results differed significantly under a paired t-test with $p < 0.05$.

5. Conclusion & Future Work

The similarity measure that we defined was applied in a clustering task and we gave an example of discovered interesting vessel behavior that is a combination of both raw trajectories and geographical information. We also used the measure in classification to predict vessel types where the combined similarity showed the best performance in terms of classification accuracy. We also plan to apply the measure in the task of outlier detection to discover strange vessel behavior.

References

de Vries, G., & van Someren, M. (2010). Clustering vessel trajectories with alignment kernels under trajectory compression. *ECML/PKDD (1)* (pp. 296–311). Springer.

Dhillon, I. S., Guan, Y., & Kulis, B. (2007). Weighted graph cuts without eigenvectors – a multilevel approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*, 1944–1957.

Gudmundsson, J., Katajainen, J., Merrick, D., Ong, C., & Wolle, T. (2009). Compressing spatio-temporal trajectories. *Computational geometry*, *42*, 825–841.

Spaccapietra, S., Parent, C., Damiani, M. L., de Macêdo, J. A. F., Porto, F., & Vangenot, C. (2008). A conceptual view on trajectories. *Data & knowledge engineering*, *65*, 126–146.

van Hage, W. R., Wielemaker, J., & Schreiber, G. (2010). The space package: Tight integration between space and semantics. *T. GIS*, *14*, 131–146.