EXperimental Learning

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Big Data and Social Analytics certificate course

© 2016 MIT / GetSmarter All Rights Reserved (not authorized for commercial use)

MODULE 4 UNIT 4
Video 8 Transcript

MIT BDA Module 4 Unit 4 Video 8 Transcript

Speaker key

XD: Xiaowen Dong

HY: Hapyak

00:00:00

XD: In this video, we will talk about an emerging research field in graph-based data processing, namely graph signal processing, which utilizes signal processing tools to analyze graph-structured data.

I will first give you some motivations of our studies, review some basic concepts in the field and then mainly focus on one practical example of how we can apply graph signal processing tools to solve real-world problems.

HY: Why graph signal processing

XD: In previous videos, we mainly focused on the edge structure of the graph, namely how the vertices are connected in the graph and how we can use such structure to partition the graph.

In addition to edge structure, it is often the case that we also have information associated with the vertices of the graph and here are a few examples. In the first example, we have a road network in US freeways. In this case, the vertices of the graph represent different locations in the freeways, which are connected by edges if their geographical distances are smaller than the threshold. The information defined on the vertices of the graph are the amount of traffic delays in the freeways.

00:01:16

In the second example, we have another geographical graph in the city center of London where the vertices represent different locations and, again, they are connected by edges if they are sufficiently close by. The information defined on the vertices, in this case, are the number of distinct users who have taken photos and uploaded on Flickr.

In the final example, we have a social network graph on Twitter where the vertices obviously represent the Twitter users and, if one user follows the other, or being followed, then there is an edge connecting them. In this case, the information defined on the vertices are the number of Tweets each person has tweeted that contain a specific keyword.

In addition, in all of these cases, information defined on the vertices are influenced by the structure of the graph. We can imagine that traffic delays may depend on the topology of the road network, the number of people who take photos are going to be influenced by the structures of the city, and activities of Twitter users are obviously going to be influenced by their social friendship.

© 2016 MIT / GetSmarter All Rights Reserved (not authorized for commercial use)

00:02:18

HY: Q: The information on the vertices of the previous image of London is most likely influenced by which of the following:

a. Physical structure of the city and specific landmarks
b. The weather conditions
c. The time zone that the city falls under

The underlying structure of the city and its landmarks will influence the information defined on the vertices in this graph. Similarly, the typology would influence the graph showing traffic on freeways and the information on the graph showing twitter usage will be influenced by the connections that already exist between the users.

XD: Therefore, we would like to take into account such underlying structure when we process the information of interest.

HY: Graph signal representation

XD: Such structure of the information can be represented by graph signals where the edges in the graph capture the relationships between the vertices that represent either locations or people, in the previous examples, where the signal represents information of interest, such as traffic delays or user activities.

Mathematically, a graph signal is a function *f* that assigns a scale of value to each vertex in the graph. In this example, we have a graph signal, which is defined on a graph of nine vertices and we see that red bars represent positive values on vertices 1, 2, 3, 4. Blue bars represent negative values on vertices 6, 7, 8, 9 and there is a zero value on vertex 5. So, this is a typical example of a graph signal.

As you can see, graph signals provide nice and compact representations that take into account both information on edges and vertices. Moreover, generalization of classical signal processing tools can benefit the processing of such signals. In the following, I will give you a small example of such generalization.

HY: Frequency analysis on graphs

XD: One of the powerful analysis tools in classical signal processing is frequency-based analysis. The frequency domain representation of a signal can be obtained by applying the so-called Fourier Transform, which decomposes the signal into the frequencies that make it up. Conceptually, the decomposition is done by looking at how similar the signal is to each of a set of sinusoid functions that change at different frequencies. This provides insights into the behavior of the signal namely, the more the signal looks like a high-frequency function, the faster it tends to change over time.

00:04:17

Naturally, we would like to have a similar notion of frequency for graph signals which would help us study how fast the signal varies with respect to the graph topology. This is helpful, for example, when

© 2016 MIT / GetSmarter All Rights Reserved (not authorized for commercial use)

we are interested in understanding how personal preferences shift between friends in a social network. The graph Laplacian matrix, which we have mentioned in the spectral graph partitioning approach, provides such a notion of frequency on graphs.

In this figure, we show three eigenvectors of the graph Laplacian matrix which corresponds to increasing eigenvalues. We see that chi 0, which is the eigenvector corresponding to the first eigenvalue, is basically a constant signal on the graph without any variation. The next one, chi 1, sees increasing variations, but such variations are smooth in a sense that there is no sharp transition of values along edges. This is no longer the case when we look at chi 50, whose values oscillate much more frequently.

These eigenvectors provide a basis which is, in one sense, similar to the sinusoid functions in the classical setting. By looking at how similar the graph signal is to each of these eigenvectors, we get a sense how fast the signal varies on the graph. In the additional notes, we will explain that this actually leads to a nice analogy to the classical Fourier transform, which we call graph Fourier transform.

00:05:42

HY: Mobility inference using Flickr data

XD: In this example, we will look at the problem of mobility inference using Flickr data. Flickr is a popular photo-sharing platform on the internet and here you can see the interface that shows the photos that have been taken by my friend. When people take photos and upload, it is often the case that the photos come with a geotag that indicates exactly where the photo was taken. And here we see a graph signal that is constructed using the Flickr data.

In this case, we have a geographical graph where the vertices represents different locations in the city center of London. We connect every pair of vertices if the distance between them is smaller than the threshold. The signal defined on the vertices of the graph is the number of distinct Flickr users who have ever taken photos in that exact location.

The values of the signal is indicated by the color on the vertices namely, the brighter the colors, the higher the value of the signal, the more people who have taken photo in that location. The signal that we have here is based on aggregated information over a period of two years and a half. It is not difficult to imagine that the values of the signal in this case are correlated with the structures of the graph, because we can imagine that close by locations usually attract a similar number of people taking photos there.

00:07:00

To take a closer look, we can zoom in to four different regions in the city to see how the signals look differently in these local regions. For example, it is clear to see that the local graph signal that represents Buckingham Palace is very different from the one that represents a bridge. Intuitively, these differences come from the differences in the mobility patterns of the people who visit these areas. Therefore, the question that we want to ask is that, by just looking at such an aggregated signal, can we try to infer the different mobility patterns in these different regions?

© 2016 MIT / GetSmarter All Rights Reserved (not authorized for commercial use)

In this case, if we think of information in the local regions as truncated signals defined on the smaller graphs, then actually, the problem is to try to distinguish these different truncated graph signals according to different information patterns, in this case, the mobility patterns. If we take a closer look at these four truncated graph signals, then actually, they correspond to different mobility patterns that we expect.

The first signal may correspond to a random movement pattern in places such as an open square, without significant landmark, and people just move around randomly. Where the second signal may correspond to a spreading pattern, for example in places near the terminals of public transport stations, where people get out of the tube stations or bus stations and move towards their landmarks nearby. The third signal may correspond to a gathering pattern where there is a significant landmark, so that people just go there directly, take a photo and leave. The fourth signal may correspond to a bi-directional pattern due to a geographical constraint, for example people move bi-directionally along streets or bridges.

00:08:40

So, our approach to mobility inference is as follows. Based on these four expected patterns, we create synthetic signals by simulating synthetic trajectories of users using random walks. We then compare these synthetic signals with the real-world signals so that we can match what we observe in real world to one of the synthetic patterns that we have in mind, because both the synthetic signals and the real-world signals are actually signals defined on a geographical graph.

Therefore, we need to extract efficient features for such a matching and, in our approach, we use wavelet transforms defined on graphs to extract such features. Once we have the features, we can use machine-learning algorithms to train the classifier and apply them to the real world signals, which will help us classify each of the real world signals into one of the expected patterns.

The result of such a matching, or inference, is shown in this figure. In this figure, the blue labels represent vertices that have too few neighbors so that their neighborhoods are not of significant interest. The cyan, yellow, orange and red labels represent vertices whose neighborhoods are classified as those having the random, spreading, gathering and bi-directional mobility patterns respectively. As we can see, the red vertices are mostly streets or bridges over the river, which corresponds well to the assumption we made in the bi-directional pattern. The orange vertices clearly represent landmarks or locations of significant interest whose neighborhoods fit well in the gathering pattern. The yellow vertices stand for regions similar to the ones in the spreading pattern where we expect movement of Flickr users from certain directions to the points of interest nearby. Therefore, they are usually located in the areas surrounding the orange vertices. Finally, the cyan vertices represent the areas where the number of Flickr users are quite uniform. For example, squares with no particularly interesting landmarks.

00:10:40

These results show that the labels obtained by the classifier based on the features that we have are indeed meaningful. In the example, we are able to infer mobility patterns for only a coarse and aggregated signal, without looking at the empirical traces from the actual Flickr users. This is achieved

© 2016 MIT / GetSmarter All Rights Reserved (not authorized for commercial use)

by matching the real-world graph signals with the synthetic ones using features extracted by applying graph signal processing tools.

So, in this video, we have introduced some basic concepts in graph signal processing and, if you find it interesting, there are more materials in the additional notes.

© 2016 MIT / GetSmarter All Rights Reserved (not authorized for commercial use)