Slides Based on
http://snap.stanford.edu/class/cs224w-2015/recitation/CS224w-2015-Snappy.pdf

Introduction to SNAP.PY

HTTP://SNAP.STANFORD.EDU/SNAPPY/

Zhedi Liu

September 30, 2016

What is SNAP?

- **S**tanford **N**etwork **A**nalysis **P**latform (SNAP) is a general purpose, high-performance system for analysis and manipulation of large networks.
  - http://snap.stanford.edu
  - Scales to massive networks with hundreds of millions of nodes and billions of edge
- SNAP Software
  - Snap.py for Python; SNAP C++
- SNAP Datasets
  - Over 70 network datasets
  - http://snap.stanford.edu/data/

Snap.py Resources

- **Prebuilt packages** available for Mac OS X, Windows, Linux
  - http://snap.stanford.edu/snappy/index.html
- **Snap.py documentation**: **(IMPORTANT!!!)**
  - http://snap.stanford.edu/snappy/doc/index.html
  - Quick introduction, Tutorial, Reference Manual
- **SNAP user mailing list**
  - http://groups.google.com/group/snap-discuss
- **Developer resources**
  - Software available as open source under BDS license
  - GitHub repository
  - https://github.com/snap-stanford/snap-python

SNAP Network Datasets

- **Collection of over 70 social network datasets**
  - http://snap.stanford.edu/data

Installing Snap.py

- **Requires Python 2.7**
  - http://www.python.org/
- **Download the Snap.py for your platform**
  - http://snap.stanford.edu/snappy
- **Installation**
  - Follow instructions here: http://snap.stanford.edu/snappy/index.html
  - (sudo) python setup.py install
- **Problems?**
  - Refer to our troubleshooting guide: https://docs.google.com/document/d/1iuFKw0mS5GsrVj7T7opXDY-qE8fbtd6HTJBZhDYeE3Q/edit
  - Post or looking at existing posts on Piazza

Snap.py: Important

- The most important step:
  - **Import the snap module!!!**

```
$ python
>>> import snap
```

Snap.py Tutorial

- **On the Web:**
  - http://snap.stanford.edu/snappy/doc/tutorial/index-tut.html
- **We will cover:**
  - Basic Snap.py data types
  - Vectors, hash tables and pairs
  - Basic graphs types
  - Graph creation
  - Adding and traversing nodes/edges
  - Useful functions to know for HW0

Basic Types & Vector Types

- **Basic types in SNAP are `TInt`, `TFlt`, and `TStr`**
  - Corresponding to Python types **`int`**, **`float`** and **`str`**
- **Vector Types**
  - **Naming convention**: `T<value_type>V`
  - **Examples**: TIntV, TFltV, TStrV
  - **Common operations:**
    - **`Add(<value>)`**: append a value at the end
    - **`Len()`**: vector size
    - **`[<index>]`**: get or set a value of an existing element
    - **`for i in V`**: iteration over the elements

Vector Example

```
v = snap.TIntV()

v.Add(1)
v.Add(2)
v.Add(3)
v.Add(4)
v.Add(5)

print v.Len()

print v[3]
v[3] = 2*v[2]
print v[3]

for item in v:
    print item
for i in range(0, v.Len()):
    print i, v[i]
```

Create an empty vector

Add elements

Print vector size

Get and set element value

Print vector elements

Hash Table Types

- **A set of (key, value) pairs**
  - Keys must be of the same types
  - Values must be of the same types
  - Value type can be different from the key type
- **Naming convention**: T<key_type><value_type>H
- **Examples**: TIntStrH, TIntFltH, TStrIntH
- **Common operations**:
  - **`[<key>]`**: add a new value or get or set an existing value
  - **`Len()`**: hash table size
  - **`for k in H`**: iteration over keys

Hash Table Example

```
h = snap.TIntStrH()

h[5] = "apple"
h[3] = "tomato"
h[9] = "orange"
h[6] = "banana"
h[1] = "apricot"

print h.Len()

print "h[3] =", h[3]

h[3] = "peach"
print "h[3] =", h[3]

for key in h:
    print key, h[key]
```

Create an empty table

Add elements

Print table size

Get element value

Set element value

Print table elements

Pair Types

- **A pair of (value1, value2)**
  - Type of value1 can be different from type of value2
- **Naming convention**: **`T<type1><type2>Pr`**
- **Examples**: TIntStrPr, TIntFltPr, TStrIntPr
- **Common operations**:
  - **`GetVal1`**: get value1
  - **`GetVal2`**: get value2

Pair Example

```
>>> p = snap.TIntStrPr(1,"one")

>>> print p.GetVal1()
1
>>> print p.GetVal2()
one
```

Create a pair

Print pair values

Basic Graph Classes

- **Graphs**
  - **TUNGraph**: undirected graph
  - **TNGraph**: directed graph
  - **TNEANet**: multigraph with attributes on nodes and edges

Graph Creation Example

```
G1 = snap.TNGraph.New()

G1.AddNode(1)
G1.AddNode(5)
G1.AddNode(12)

G1.AddEdge(1,5)
G1.AddEdge(5,1)
G1.AddEdge(5,12)

G2 = snap.TUNGraph.New()
N1 = snap.TNEANet.New()
```

Directed graph

Add nodes before adding edges

Undirected graph, directed network

G1

Graph Traversal Example

Node traversal

```
for NI in G1.Nodes():
    print "node id %d, out-degree %d, in-degree %d"
      % (NI.GetId(), NI.GetOutDeg(), NI.GetInDeg())
```

Edge traversal

```
for EI in G1.Edges():
    print "(%d, %d)" % (EI.GetSrcNId(), EI.GetDstNId())
```

Edge traversal by nodes

```
for NI in G1.Nodes():
    for DstNId in NI.GetOutEdges():
        print "(%d %d)" % (NI.GetId(), DstNId)
```

Graph Saving and Loading

Save text

```
snap.SaveEdgeList(G4, "test.txt", "List of edges")
```

Load text

```
G5 = snap.LoadEdgeList(snap.PNGraph,"test.txt",0,1)
```

Save binary

```
FOut = snap.TFOut("test.graph")
G2.Save(FOut)
FOut.Flush()
```

Load binary

```
FIn = snap.TFIn("test.graph")
G4 = snap.TNGraph.Load(FIn)
```

Loading Text File

Example file: wiki-Vote.txt

- Download from http://snap.stanford.edu/data

```
# Directed graph: wiki-Vote.txt
# Nodes: 7115 Edges: 103689
# FromNodeId    ToNodeId
0       1
0       2
0       3
0       4
0       5
2       6
…
```

```
LoadEdgeList(PGraph, InFNm, SrcColId, DstColId, Separator)
G = snap.LoadEdgeList(snap.PNGraph, "wiki-Vote.txt", 0, 1)
```

Useful Functions to Know: `G.Nodes()` & `G.Edges()`

- Get a generator for all nodes in G
  - http://snap.stanford.edu/snappy/doc/reference/graphs.html?highlight=nodes()
- Get a generator for all edges in G
  - http://snap.stanford.edu/snappy/doc/reference/graphs.html?highlight=edges()
- Example Usage

```
for node in G.Nodes()
for edge in G.Edges()
```

Useful Functions to Know: `G.GetNodes()` & `G.GetEdges()`

- Get the total number of nodes in G
  - http://snap.stanford.edu/snappy/doc/reference/graphs.html?highlight=getnodes
- Get the total number of edges in G
  - http://snap.stanford.edu/snappy/doc/reference/graphs.html?highlight=getedges
- Example Usage

```
G = snap.LoadEdgeList(snap.PNGraph, "wiki-Vote.txt", 0, 1)
print "G: Nodes %d, Edges %d" % (G.GetNodes(), G.GetEdges())
```

Useful Functions to Know: `CntSelfEdges(G)` & `CntUniqDirEdges(G)`

- Get the total number of self edges
  - http://snap.stanford.edu/snappy/doc/reference/CntSelfEdges.html
  - Example Usage

```
Count1 = snap.CntSelfEdges(G)
print "Count of self edges is G is %d" % Count1
```

- Get the total number of unique directed edges
  - http://snap.stanford.edu/snappy/doc/reference/CntUniqDirEdges.html
  - Example Usage

```
Count2 = snap.CntUniqDirEdges(G)
print "Count of unique directed edges is %d" % Count2
```

Useful Functions to Know: `CntUniqUndirEdges(G)` & `GetInDeg()`

- Get the total number of unique undirected edges
  - http://snap.stanford.edu/snappy/doc/reference/CntUniqUndirEdges.html
  - Example Usage

```
Count3 = snap.CntUniqUndirEdges(G)
print "Count of unique undirected edges is %d" % Count3
```

- Get in degree of a node `n`:
  - http://snap.stanford.edu/snappy/doc/reference/graphs.html?highlight=getindeg
  - Example Usage

```
n.GetInDeg()
```

Useful Functions to Know: `GetOutDeg()` & `GetWccs(G, C)`

- Get out degree of a node `n`:
  - http://snap.stanford.edu/snappy/doc/reference/graphs.html?highlight=getoutdeg
  - Example Usage

```
n.GetOutDeg()
```

- Get all weakly connected components in G
  - http://snap.stanford.edu/snappy/doc/reference/GetWccs.html
  - Example Usage

```
Components = snap.TCnComV()
snap.GetWccs(G, Components)
for CnCom in Components:
    print "Size of component: %d" % CnCom.Len()
```

Useful Functions to Know: `GetMxWcc(G)` & `GetPageRank(G, P)`

- Get largest weakly connected component in G:
  - http://snap.stanford.edu/snappy/doc/reference/GetMxWcc.html
  - Example Usage

```
MxWcc = snap.GetMxWcc(G)
for EI in MxWcc.Edges():
        print "edge: (%d, %d)" % (EI.GetSrcNId(), EI.GetDstNId())
```

- Get the PageRank score of every node in G
  - http://snap.stanford.edu/snappy/doc/reference/GetPageRank.html
  - Example Usage

```
PRankH = snap.TIntFltH()
snap.GetPageRank(G, PRankH)
sorted_PRankH = sorted(PRankH, key = lambda key: PRankH[key],
                                reverse = True)
```

Useful Functions to Know: `GetHits(G, H, A)`

- Get the Hubs and Authorities score of every node in G:
  - http://snap.stanford.edu/snappy/doc/reference/GetHits.html?highlight=gethits
  - Example Usage

```
NIdHubH = snap.TIntFltH()
NIdAuthH = snap.TIntFltH()
snap.GetHits(G, NIdHubH, NIdAuthH)
sortedByAuth = sorted(NIdAuthH, key = lambda key: NIdAuthH[key], reverse = True)
sortedByHub = sorted(NIdHubH, key = lambda key: NIdHubH[key], reverse = True)
```

Thank You !