txt2pajek: Creating Pajek Files from Text Files

Authors: Jürgen Pfeffer, Andrej Mrvar, Vladimir Batagelj

December, 2015
CMU-ISR-15-???

Institute for Software Research
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Technical Report

Keywords: Pajek, network analysis, network data, text files

Abstract

Pajek is a software tool for analysis and visualization of large networks and has been under constant development since 1996. In 2004, the first version of txt2pajek was released to assist scientists in all areas to create Pajek readable .net files from raw text files. In the following years several updates have been released. Now we present a new version that incorporates recent advancements in Pajek and more complex network structures (e.g. handling of Unicode data, multiplex networks, vectors, partitions). This technical report describes the different options in txt2pajek and can also be seen as an introduction to creating Pajek network files.

Table of Contents

1 Introduction

Pajek[1] (Nooy, Mrvar, & Batagelj, 2011) is a software tool for analyzing social networks. Pajek was developed to support the analysis of very large networks (Batagelj & Mrvar, 1998), as well as the visualization of networks (Batagelj & Mrvar, 2003). Pajek is used by thousands of network researchers in many countries. Recently, textbooks in Japanese (Nooy, Mrvar, Batagelj, & 安田, 2009) and Chinese (Nooy, Mrvar, & Batagelj, 2012) were published.

Researchers are used to handling their data in statistical tool, spreadsheet programs, or databases. A crucial pre-condition for analyzing networks is to convert network data into network files that can be read by network tools. This is the purpose of txt2Pajek. In this tech report, we describe how to use txt2Pajek to convert data stored in text files to Pajek network files. We first review the basics of the Pajek data format and the format of the text files that can be used as input for txt2Pajek. Then, we describe the basic process of converting text files to Pajek files by using txt2Pajek. Finally, advanced options are explored to create Pajek files with additional information (e.g. link labels, temporal information, multi-relational networks, vectors, partitions).

1.1 Pajek Data format

Pajek works with a rather easy and straightforward approach in handling data files. It is important to know that all Pajek files are plain text files that can be read with any text tools. However, you should not use "advanced" text tools like Microsoft Word and the like that add formatting information to the text file. Instead, use regular text editors (e.g. Textpad, or BabelPad for Unicode files).

Most of your activities in Pajek will result in one or more of these three data objects: networks, partitions, and vectors. We do not discuss other data objects in this document but you can find more about all data objects in the Pajek manual[2]. Networks, vectors, and partitions are stored in different file formats that Pajek can read and write. Figure 1 shows examples for these three file formats. In contrast to other SNA tools, Pajek stores files in plain text format. This has several advantages. First, readability; the files can be opened and modified in any text editor. Second, compatibility; files can be exchanged between Pajek and other tools quickly and in both directions. Third, it is easy to create files that Pajek can read from other tools.

`*Vertices 4`
`1 "George"`
`2 "Susan"`
`3 "John"`
`4 "Sarah"`
`*Edges`
`1 2`
`2 3`
	`*Vertices 4`
	`1`
	`2`
	`1`
	`2`
		`*Vertices 4`
		`0.25`
		`0.50`
		`0.10`
		`0.70`

[1] In Slovenian language Pajek means spider.
[2] http://pajek.imfm.si/lib/exe/fetch.php?media=dl:pajekman.pdf

3 4		

Figure 1: Pajek file format. Left: .net network file. Center: .clu partition file. Right: .vec vector file.

1.2 Format of Text Files

txt2Pajek works with regular text files. Most tools (e.g. Microsoft Excel) or databases have the ability to export data in this format. Look for tabulator separated text files .txt or .tab. Comma separated or any other text format is possible, however, we highly recommend tab-separated files. Avoid working with advanced text processing tools (e.g. Microsoft Word) as these files have additional formatting and other meta information stored in the file. A typical text file that serves as input for txt2Pajek looks like what is shown in the left part of Figure 1. You can see three columns, two for node information and one column describing the link weights. We call this format *edge list*, as every line describes a single edge in the network. Independently from the complexity of your data, the basic form of one edge by line must be guaranteed, e.g.:

```
from    to    weight    link.color    link.type    time    etc…
```

This approach results in additional columns in the text file for additional information. That is the reason why the txt2Pajek user interface consists of many dropdown objects which are used to assign a column from the text file to a specific network attribute. In our simple example, the text file consists of three columns and four (without the header line) lines with different values. These lines can be seen in the Pajek file (Figure 1 center) and in the network picture of on the right of Figure 1. Please note that there is no definition of nodes in this text file. Nodes are implicitly defined as they are part of links.

```
From       To         Weight
Jim        John       1
John       George     3
George     Berta      4
Jim        George     3
```

```
*Vertices 4
1 "Jim"
2 "John"
3 "George"
4 "Berta"
*Edges
1 2 1
2 3 3
3 4 4
1 3 3
```

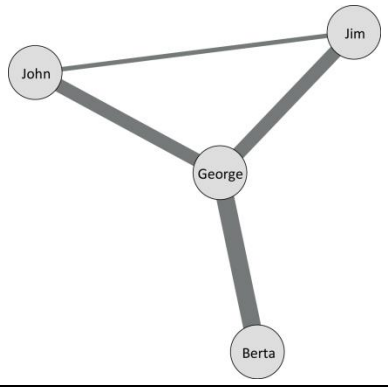

Figure 1: edgelist.txt (left) and the resulting Pajek file (center) with four nodes and four edges as well as the network visualization (right).

2 Basic Functionality

txt2Pajek 3 has more features than previous versions. To reduce the complexity of the tool, there are two layers of options (see Figure 2), basic options and advanced options. In the following we discuss the *basic options*. On top of the tool you can find three buttons.

"Run" starts the conversion process, "Info" shows tool information and the link to the related web page, "Exit" quits the program.

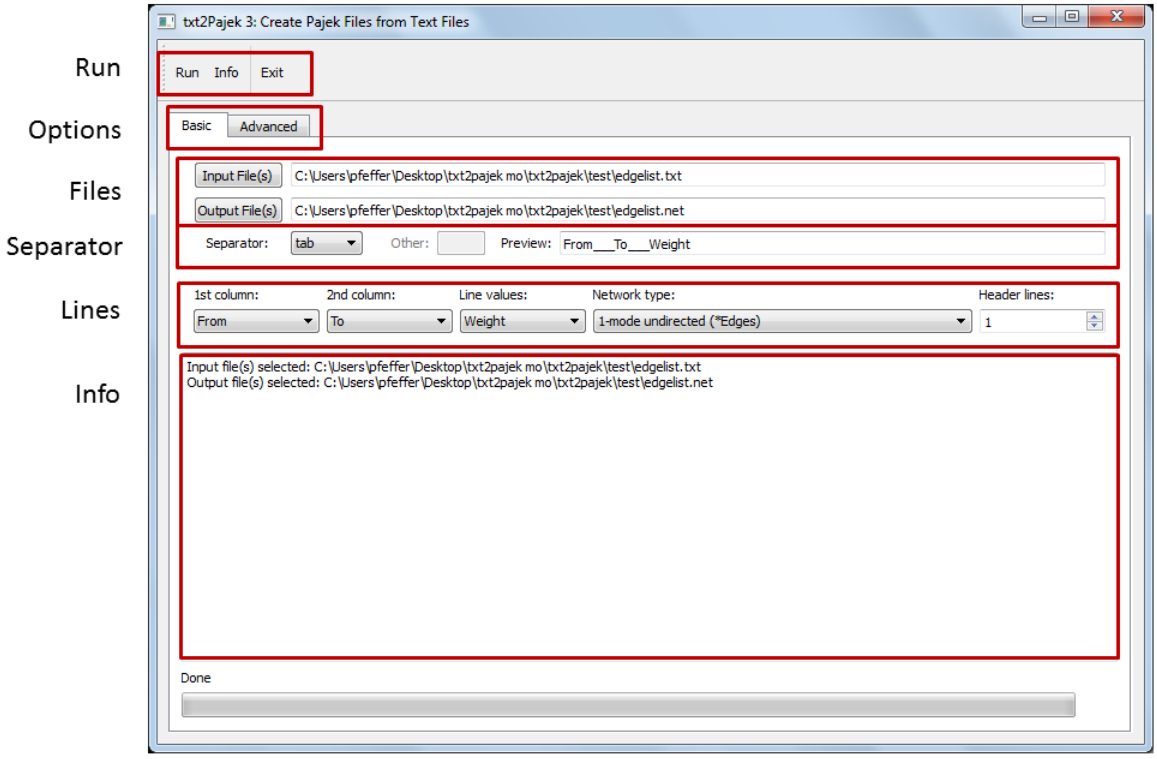

Figure 2: The basic functionalities highlighted in the txt2Pajek 3 main window.

txt2Pajek makes some decisions on handling your text files without giving you the option to change it:

- Multiple lines stay multiple lines. If your input data has multiple lines from A to B, then your .net file will have the same number of multiple lines. You can aggregate these multiple lines later in Pajek if you like.
- If there are any quotation marks, i.e., " or ', they will be removed from the text. This is important as Pajek uses quotation marks to indicate beginning and ending of text.

2.1 Files

The first thing you do when starting txt2Pajek is select an input file. You can also select multiple input files in the file open dialog by pressing the Shift or the Ctrl key while selecting files with the mouse. The output file gets set automatically to the same path and filename as the input file but with a .net extension indicating a network file in Pajek format. You can change the name of the output files manually.

2.2 Separator

Every text file needs a separator that indicates the separations of different column information. txt2Pajek offers four pre-defined separator, tabulator, comma, semi-colon, and space (blank). By selecting other, you can use any character or combination of characters as separators. However, we strongly recommend using tabulators as separators as all other characters could be part of your data. As a matter of fact, mixing values with separators is a common error, e.g.

```
Name,Country,Value
John,Austria,1
Joe,Netherlands, The,2
```

will result in an error as the additional comma in Joe's line cannot be interpreted correctly; "The" is not valid for a link value. The text field next to "Preview:" tells you whether your separator selections work or not. "___" is used to indicate the separator. For instance, if the first line in your text file is "nameA,nameB" and "comma" is selected as separator then the preview will show "nameA___nameB". If "other" is selected and "a" is added as other separator, then "n___meA,n___meB" is the result. Selecting the right separator is crucial for txt2Pajek in order to identify the columns correctly that also appear as values in the dropdown objects of the GUI.

2.3 Lines

A link in a network is described by two essential pieces information, a source and a sink node. Every link connects two nodes. The columns with these nodes are selected with the two dropdowns "1st Column" and "2nd Column". The third dropdown selects the link values. This selection is optional. If your network is not weighted (no different values for links) then "1" is selected to add the value 1 for every link in the network.

For the "Network Type" there are three different options to select. For one mode networks there are two different option, either directed or undirected. Two mode networks are undirected:

a) 1-mode undirected (*Edges):

b) 1-mode directed (*Arcs):

c) 2-mode undirected (*Edges):

The option "Header Lines" is used to tell txt2Pajek how many lines from top of the text file should be ignored because they include header information and not network

information. For the example in Figure 1, we would select “1” as the first line should not be included to create the network file.

2.4 Info

The info window can be seen as protocol. Information about the created networks and error messages are written here.

3 Advanced Options

Advanced options can be found on the “Advanced” tab. The options describe additional line information, optional partition and vector files as well as additional options related to the text file or the network type.

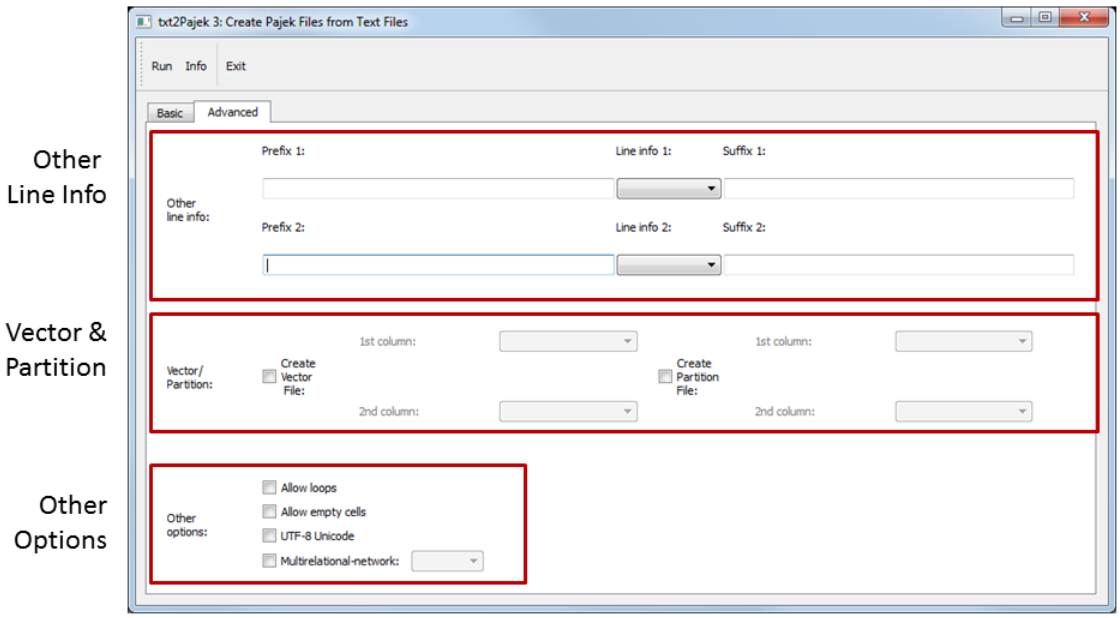

Figure 3: Overview of advance features.

3.1 Other Line Info

With “other line info” some advanced options are possible. The underlying logic is the following:

```
prefix    line info    suffix
```

You can use this to code any additional information for lines. For instance, if <column> represents the column selections of the dropdown of line info 1 or line info 2, then these are possible examples for prefix/line info/postfix combinations:

`c <column>`	… for line colors
`"<column>"`	… for line labels

`[<columns>]` … for temporal information

3.2 Vector/Partition Files

One of the new features of txt2Pajek 3 are the options to create vector and partition files during the network creation process. In Pajek, partitions are used to split nodes into non-overlapping groups based on a nominal variable, e.g. gender, nationality, department, etc. Vectors are used to store quantitative information of nodes, e.g. salary, size, etc. as well as centrality metrics. The edge list logic of txt2Pajek has two implications for incorporating vector and partition information (see Figure 4). First, we need two columns for the information, one for each node column. Second, vector and partition information needs to be redundant as normally nodes occur in more than one line.

Of course, this can be a painful task if you prepare your text file by hand, but when the data is exported from a database, this should not be a problem.

```
nameA    genderA   nameB   genderB
George   1         Susan   2
Susan    2         John    1
John     1         Sarah   2
```

Figure 4: Text file to create a network and a partition file.

3.3 Allow Loops

A loop is a link from a node to itself. The example in Figure 5 shows that "wei" is connected to "wei". Loops are special in network analysis and it is important to be aware of whether your data has loops or not – creating a network with loops should be an explicit decision. Consequently, we added the option "Allow Loops" that is not selected by default. txt2Pajek always checks for loops and you will get an error message if your data contains a loop and this option is unselected.

```
nameA,nameB
wei,wei
george,paul
paul,joa
```

Figure 5: Illustrating loops in text files.

3.4 Allow Empty Cells

Empty cells can be the result of data errors. If your data has an empty cell the conversion process stops and no file is created. But there are cases for which empty cells are tolerated or even necessary. The main purpose of empty cells is to include nodes to the networks that have no links (isolates). As the link list format just includes nodes that are

part of links, an empty cell in the link list is the only way of including isolates to a network file. In Figure 6 you can see that “Marc” is not connected to a city. The option 'allow empty cells' creates a “*UNKNOWN*” vertex (in case of 2-mode: “*UNKNOWN1*” and “*UNKNOWN2*”). You can delete these nodes later in Pajek, then Marc will be an isolate.

```
Name      City
George    Pittsburgh
John
John      Paris
Sue       Paris
Marc
          Paris
Sue       Georgia
```

```
*Vertices 9 5
1 "George"
2 "John"
3 "Sue"
4 "Marc"
5 "***UNKNOWN1***"
6 "Pittsburgh"
7 "***UNKNOWN2***"
8 "Paris"
9 "Georgia"
*Edges
1 6 1
2 7 1
2 8 1
3 8 1
4 7 1
5 8 1
3 9 1
```

Figure 6: Empty cells in the text files (left) and the resulting network file (right).

3.5 UTF-8 Unicode

UTF-8 is a code format that incorporates special characters from many languages (including Japanese and Chinese). Another term that is used instead of UTF-8 is “Unicode”.

Regular text editors often do not use Unicode. If you open a UTF-8 file in those editors you get a warning message that UTF-8 characters are getting destroyed. There are special Unicode editors that overcome this problem, e.g. BabelPad, XPad, jEdit. The most important thing is that your editor should be capable of saving text files as “UFT-8 with Byte Order Mark” or BOM. This is the Unicode format of Pajek.

3.6 Multi-Relational Networks

One network can have different groups of lines. For instance, two nodes can be “friends” while two other nodes are “colleagues”. You can create different networks for every type of links but you can also create one network that incorporates different link types. This is called a multi-relational network in Pajek. For txt2Pajek it is necessary to code this information in an additional column (see Figure 7).

```
nameA     nameB     relation
George    Susan     friends
```

```
*Vertices 6
1 "George"
```

John George colleague Susan John friends John Sarah friends Victor Jim colleague Jim John no relation	2 "Susan" 3 "John" 4 "Sarah" 5 "Victor" 6 "Jim" *Edges :1 "colleague" 3 1 1 5 6 1 *Edges :2 "no relation" 6 3 1 *Edges :3 "friends" 1 2 1 2 3 1 3 4 1

Figure 7: Multi-relational network. Text file (left) and resulting Pajek network file (right).

4 CompBio

There are a couple of new features in txt2Pajek related the Computational Biology applications.

4.1 MISTIC Mutual Information File

Multiple Sequence Alignment (MSA) …
MISTIC Mutual Information (MI)
(Woods & Pfeffer, 2015)

5 Acknowledgements

Many people have been using txt2pajek in the last 10 years. Some of them gave important feedback that helped to further develop the tool. We would like to thank all of them and especially the large Pajek community. Weiqi Cai from Carnegie Mellon University was essential in converting the tool to Python, developing the GUI, and compiling the runtime .exe file.

6 References

Batagelj, V., & Mrvar, A. (1998). Pajek: A Program for Large Network Analysis. *Connections, 21*(2), 47–57.

Batagelj, V., & Mrvar, A. (2003). Pajek - analysis and visualization of large networks. In M. Juenger & P. Mutzel (Eds.), *Graph Drawing Software* (pp. 77–103). Berlin: Springer.

Nooy, W. de, Mrvar, A., & Batagelj, V. (2011). *Exploratory social network analysis with Pajek*. England; New York: Cambridge University Press.

Nooy, W. de, Mrvar, A., & Batagelj, V. (2012). 蜘蛛：社会网络分析技术. Beijing: World Publishing Corporation.

Nooy, W. de, Mrvar, A., Batagelj, V., & 安田雪. (2009). *Pajek を活用した社会ネットワーク分析*. Tokyo: Tokyo Denki University Press.

Woods, K. N., & Pfeffer, J. (2015). Using THz Spectroscopy, Evolutionary Network Analysis Methods, and MD Simulation to Map the Evolution of Allosteric Communication Pathways in c-Type Lysozymes. *Molecular Biology and Evolution*, msv178. http://doi.org/10.1093/molbev/msv178