Graphic information retrieval system on Stanford Encyclopedia of Philosophy with latent semantic indexing

Author: Joshua Wang

Program can be accessed at:
https://dl.dropboxusercontent.com/u/100310497/SEP_IR_LSI.rar

1. Motivation:
   Stanford Encyclopedia of Philosophy (SEP) is a well-known resource of philosophy, which already includes an information retrieval system. The system is useful if the query hits the title of a document. However, if it does not, the result may be too broad for the user to understand the topic he searched to choose the right entry (an example is "semantic" for the system). Therefore, instead of returning results entry by entry, this system helps the user have a better understanding of the topic by returning the structure of the documents relevant to the query, where relevancy is defined with latent semantic indexing (LSI).[1]
2. Instruction:
   1. Download the program at
      https://dl.dropboxusercontent.com/u/100310497/SEP_IR_LSI.rar
   2. Make sure *"SU_file.npy"*, *"Vt_file.npy"* , *"SEP_re_indexlist"*, *"termtoindex"* are in the same directory as *GUI.exe*
   3. Click on *"GUI.exe"*, a GUI will prompt.
   4. The program supports three modes
      (1) LSI: the resultant LSI search engine proposed in the project.
      Illustration: Each node represents a document, whose size represents its cosine similarity to the query; If the similarity between documents is greater than 0.5, an edge will appear between them, and shorter edge represents higher cosine similarity.
      (2) Website: the graphic result of the SEP website's built-in search engine. (Must have Internet access to use this mode)
      (3) Joint: The result of mode (1) filtered by mode (2). Also, the evaluation (F scores on both nodes and edges, with website's result as ground truth) will be shown in the information text frame.
   5. After entering the query and the maximum number of nodes for return, and choosing the mode for searching, click search button. The result will appear in

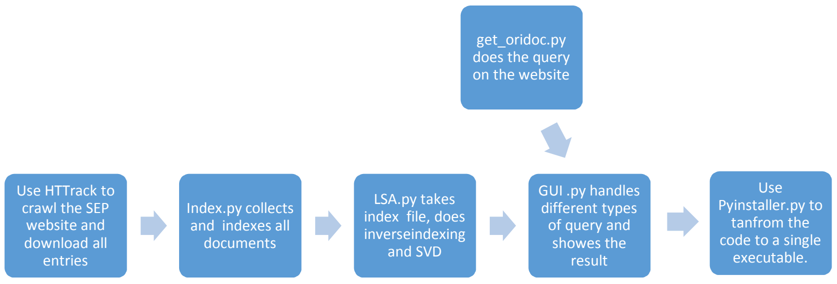
[1] Reference to Wikipedia: http://en.wikipedia.org/wiki/Latent_semantic_indexing

the canvas. Click on the node in the canvas to see the original webpage (must have Internet access to use this function)

3. Development environment:
   Application: Sublime text2 with python 2.7 64 bits
   Operating System: Windows 8
   Processor: Intel i7-4702MQ
   Memory: 8GB DDR3

4. Tools/Data:
   a. All Data are downloaded from SEP (URL: http://plato.stanford.edu/) with the tool HTTrack (URL: http://www.httrack.com/).
   b. The programming language is Python 2.7 (64 bits). With external libraries: (Note that some are unofficial because there is no official version for 64 bits python)
      (i) To do matrix operations
      numpy: http://www.lfd.uci.edu/~gohlke/pythonlibs/#numpy
      scipy: http://www.scipy.org/scipylib/download.html
      (ii) To manipulate and draw the graph
      networkx: http://networkx.github.io/
      matplotlib:http://matplotlib.org/
      dateutil: http://www.lfd.uci.edu/~gohlke/pythonlibs/#python-dateutil
      pyparsing:http://www.lfd.uci.edu/~gohlke/pythonlibs/#pyparsing
      pypi: https://pypi.python.org/pypi/six/
      (iii) To stem terms:
      Nltk: http://nltk.org/install.html
      (iv) To make the executable:
      Pyinsatller: http://www.pyinstaller.org/

5. System Description:
   Flowchart:


get_oridoc.py does the query on the website
Use HTTrack to crawl the SEP website and download all entries
Index.py collects and indexes all documents
LSA.py takes index file, does inverseindexing and SVD
GUI .py handles different types of query and showes the result
Use Pyinstaller.py to tanfrom the code to a single executable.

1. Use HTTrack to download all document in the website of SEP. All documents are under http://plato.stanford.edu/entries/.
2. Filtered out documents that do not have contents. Index.py read in all 1395 documents, with nltk, terms are lowercased and stemmed before indexing.
3. LSA.py takes index file as input, creates an inverse index, normalize, and use it to do SVD with numpy and scipy. Total number of terms is 168885. The concept space has dimension 200.
   Reference of doing LSI: http://www.puffinwarellc.com/index.php/news-and-articles/articles/33-latent-semantic-analysis-tutorial.html?showall=1
4. get_oridoc.py do the search on the website, return hit entries and links between the hit entries and entries in the "related entries" section.
5. GUI.py handle users' queries as in the instruction section.
   In LSI mode, Queries are transformed into a tf vector, and multiply $S^{-1}U^{T}$ matrix, then compare with documents in concept space. (V matrix). Documents are also compared in concept space.

   Reference for the codes:
   Overall structure:
   http://matplotlib.org/examples/user_interfaces/embedding_in_tk.html
   Click-on graph event:
   http://stackoverflow.com/questions/12894985/make-a-click-able-graph-by-networkx
   Handle graph: networkx
   http://networkx.github.io/

5. Evaluation:

1. About F score:

Some queries with good node F scores are as follow:

Query (maximum node number)	Node’s F score	Edge’s F score
Kant (30)	0.74	0
Turing (10)	0.9	0
Daoism (10)	0.7	0.15
Stoic (20)	0.7	0.77
Fichte (20)	0.6	0

Some queries with bad node F score are as follow:

Query (maximum node number)	Node’s F score	Edge’s F score
Existentialism (20)	0.05	0
Semantics(20)	0.25	0
Socialism (20)	0 (note: mode 1 and 2 both have 20 results)	0
Greek (20)	0.2	0
Tautology (20)	0.1	0

Following are the F scores of the names on the Butler Library in Columbia University. The names commonly regarded as philosopher have maximum number of nodes of 100, else 30.

Query (maximum node number)	Node’s F score	Edge’s F score
Homer (30)	0.43	0
Herodotus (30)	0.12	0
Sophocles (30)	0.13	0
Plato (100)	0.43	0.24
Aristotle(100)	0.61	0.15
Demosthenes (30)	0.05	0
Cicero (100)	0.23	0.31
Vergil (30)	0.05	0

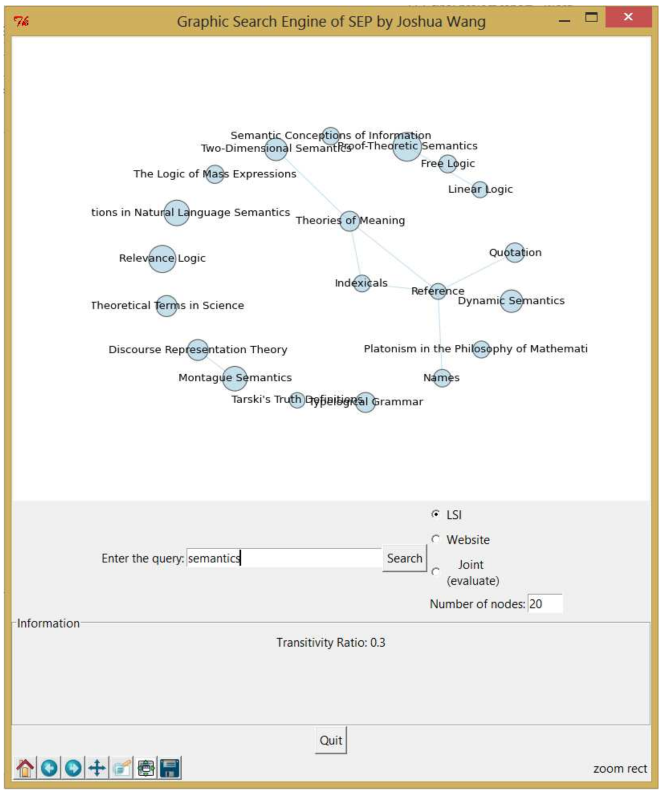
In general, queries with specific philosophy content (e.g. a philosopher) will have a better node F score; on the other hand, if the query is general (e.g. –ism) or a specific non-philosopher, the F score will be low.
The edge's F score is consistently low because the links provided by the website are few. Often the result in mode 2 does not contain any edge (an example is "Turing"), so it will have poor accuracy. But as the goal of the project is to let users have a better understanding of the topic with LSI, this is the price of the goal.

2. About program.

Originally the GUI loaded the whole indexfile and three matrices of product of SVD, which takes minutes in the development environment. A modification is made to have GUI load only the V and $S^{-1}U^T$ matrices (the latter was computed in LSA.py already) and the reduced index which contains only URL and title, as these are the information the searching really required. It reduced the loading time to a few seconds and the size of the program from 1.4 GB to about 340 MB.

6. Demonstration

Following are the outputs of some queries with interesting result:


Graphic Search Engine of SEP by Joshua Wang
Semantic Conceptions of Information
Two-Dimensional Semantics
Proof-Theoretic Semantics
Free Logic
The Logic of Mass Expressions
Linear Logic
tions in Natural Language Semantics
Theories of Meaning
Relevance Logic
Quotation
Indexicals
Reference
Dynamic Semantics
Theoretical Terms in Science
Discourse Representation Theory
Platonism in the Philosophy of Mathemati
Montague Semantics
Names
Tarski's Truth Definitions
Typelogical Grammar
Search
Quit
zoom rect

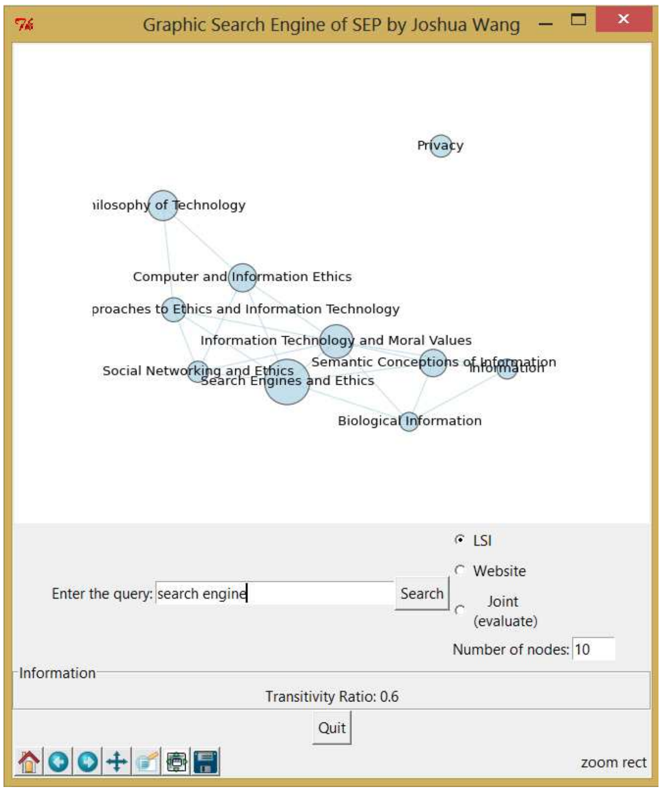
[✓ LSI] Website Joint (evaluate)

Enter the query: semantics

Number of nodes: 20

Information

Transitivity Ratio: 0.3


Privacy
ıilosophy of Technology
Computer and Information Ethics
proaches to Ethics and Information Technology
Information Technology and Moral Values
Semantic Conceptions of Information
Information
Social Networking and Ethics
Search Engines and Ethics
Biological Information
zoom rect

Graphic Search Engine of SEP by Joshua Wang

[✓ LSI] Website Joint (evaluate)

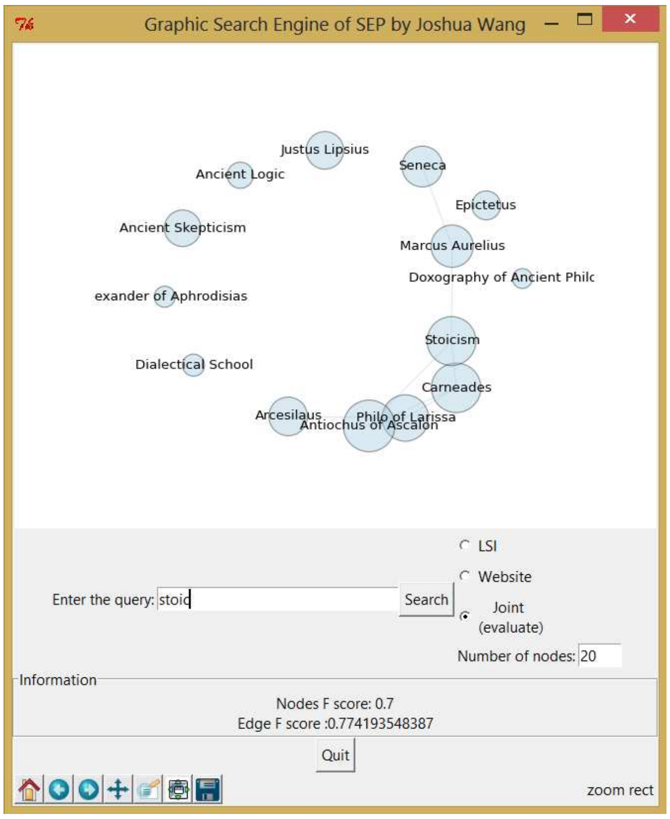
Enter the query: search engine Search

Number of nodes: 10

Information

Transitivity Ratio: 0.6

Quit



Graphic Search Engine of SEP by Joshua Wang

LSI Website [✓ Joint (evaluate)]

Enter the query: stoic Search

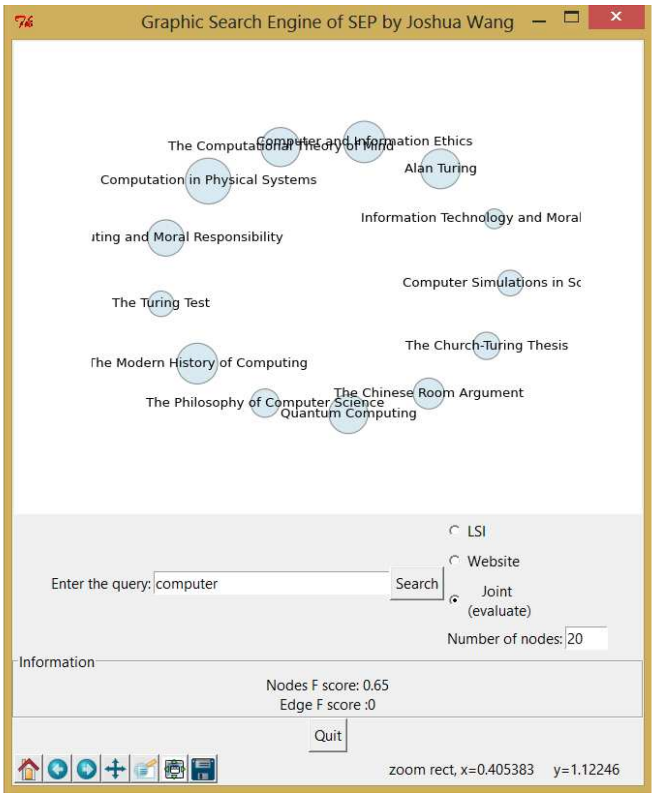
Number of nodes: 20

Information

Nodes F score: 0.7

Edge F score :0.774193548387

Quit


Graphic Search Engine of SEP by Joshua Wang
The Computational Theory of Mind
Computer and Information Ethics
Alan Turing
Computation in Physical Systems
Information Technology and Moral
ıting and Moral Responsibility
Computer Simulations in Sc
The Turing Test
The Church-Turing Thesis
The Modern History of Computing
The Chinese Room Argument
The Philosophy of Computer Science
Quantum Computing
Search
Quit

Enter the query: computer

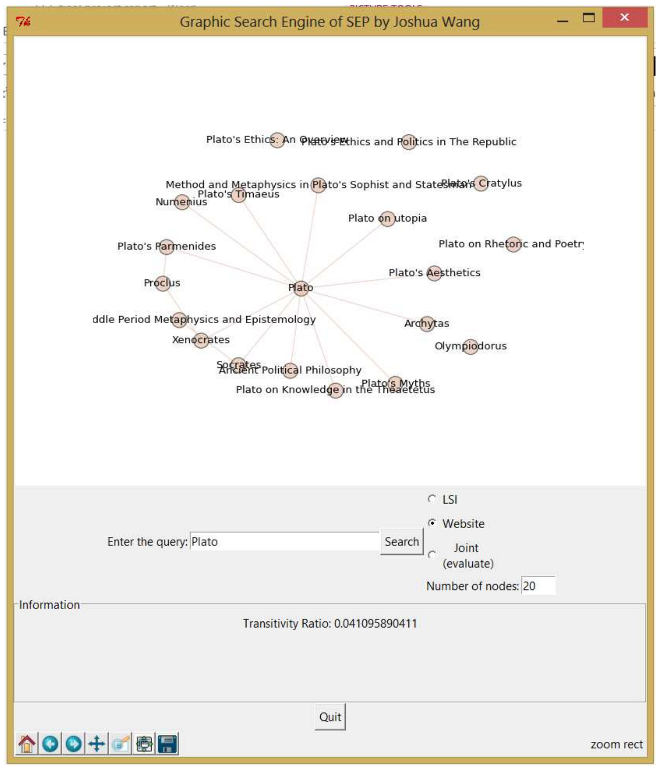
LSI
Website
[✓ Joint (evaluate)]

Number of nodes: 20

Information

Nodes F score: 0.65
Edge F score :0

zoom rect, x=0.405383 y=1.12246


Graphic Search Engine of SEP by Joshua Wang
Plato's Ethics: An Overview
Plato's Ethics and Politics in The Republic
Method and Metaphysics in Plato's Sophist and Statesman
Plato's Cratylus
Plato's Timaeus
Numenius
Plato on utopia
Plato's Parmenides
Plato on Rhetoric and Poetr
Plato's Aesthetics
Proclus
Plato
ddle Period Metaphysics and Epistemology
Archytas
Xenocrates
Olympiodorus
Socrates
Ancient Political Philosophy
Plato's Myths
Plato on Knowledge in the Theaetetus
Search
Quit

Enter the query: Plato

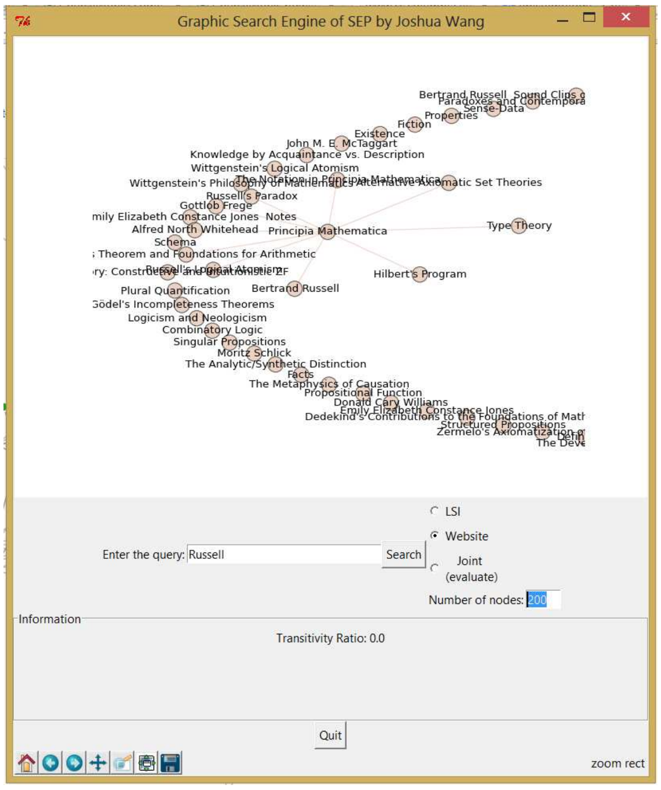
LSI
[✓ Website]
Joint (evaluate)

Number of nodes: 20

Information

Transitivity Ratio: 0.041095890411

zoom rect


Graphic Search Engine of SEP by Joshua Wang
Bertrand Russell
Sound Clips o
Paradoxes and Contempora
Sense-Data
Properties
Fiction
Existence
John M. E. McTaggart
Knowledge by Acquaintance vs. Description
Wittgenstein's Logical Atomism
Wittgenstein's Philosophy of Mathematics
The Notation in Principia Mathematica
Alternative Axiomatic Set Theories
Russell's Paradox
Gottlob Frege
mily Elizabeth Constance Jones
Notes
Alfred North Whitehead
Principia Mathematica
Type Theory
Schema
; Theorem and Foundations for Arithmetic
ry: Constructive and Intuitionistic ZF
Russell's Logical Atomism
Hilbert's Program
Plural Quantification
Bertrand Russell
Gödel's Incompleteness Theorems
Logicism and Neologicism
Combinatory Logic
Singular Propositions
Moritz Schlick
The Analytic/Synthetic Distinction
Facts
The Metaphysics of Causation
Propositional Function
Donald Cary Williams
Emily Elizabeth Constance Jones
Dedekind's Contributions to the Foundations of Math
Structured Propositions
Zermelo's Axiomatization o
Defin
The Deve
Search
Quit
zoom rect

Enter the query: Russell

( ) LSI
[✓ Website]
( ) Joint (evaluate)

Number of nodes: 200

Information

Transitivity Ratio: 0.0

(According to http://www.jacopotagliabue.it/, “Russell” is the page with the highest PageRank in the citation network of SET)