DP Course description: «Application aspects of social data processing» («Social intelligence technologies» or «Social computing»)

Annotation

Course provides practical insights for building applications involving elements of social intelligence technology involving graph analysis, semantic modeling, natural language processing and approaches for developing natural language comprehension and production systems, bot automation, social networks and transactional networks such as blockchains and marketplaces.

First block of the course is dedicated to different aspects of social data analysis and graph analysis in particular. It provides an explanation of different sorts of graph structures, applications of the graphs for semantic modeling, support of the graph processing in different systems and projects, with discussion of alternative approaches for graph representations. Also, the issue of explainability of the machine learning technology is discussed, including theoretical aspects of symbolic and sub-symbolic artificial intelligence and practical requirements to the modern machine learning solutions (GDPR, XAI, human-friendly AI).

Second block provides theoretical and practical aspects of building applications for social computing such as analytics in social networks or building applications for these networks including traditional centralized social networks such as Facebook and VKontakte as well as decentralized social and financial networks such as Ethereum, Steemit and Golos. Also, the cybernetic aspects of democracy are explained along with practical implications for building recommendations and reputation systems.

Third block of the course covers the practical aspects of developing natural language applications used to capture semantics and sentiment in online communications as well as production of texts in conversational systems such as chat-bots. Also, the notion of artificial general intelligence (AGI) is elaborated along with discussion about its feasibility and practical applications.

Major topics

Basics of graph analysis and semantic modeling, comprehension and production, social and transactional network and analytics on network data, approaches for unsupervised natural language processing, application and practical aspects of the above.

Certification format

Credit - on the basis of coursework in respect to one of the problems discussed on the lectures or mentioned in supplementary materials - including all of the following activities and submitting respective artifacts.

- Report or an abstract on one of the selected topics of the course published publicly as an essay or review on either https://steemit.com/ or https://medium.com/ or any other English-language blogging platform.
- Simple software solving one of the aspects of the problem - have to be submitted as open source under MIT license on either https://github.com/, https://bitbucket.org or https://gitlab.com/ .
- Presentation - have to be verbally presented in a seminar with slides (10-15 minutes, 7-15 slides) .

Pre-requisites

It is expected that students are familiar with the following subjects.

- Basics of Natural Language Processing
- Basics of Data Science and Machine Learning

Supplementary materials

Related videos:

https://www.youtube.com/aigents

Introduction

Anton Kolonin. Collective intelligence and Artificial intelligence - state of affairs
http://aigents.com/papers/2020/Global-Brain-and-AGI-2020.pdf
http://aigents.com/papers/2019/Social-Computing-with-Online-Communities-2019.pdf
http://aigents.com/papers/2019/AI-state-2019-en.pdf

First block (Graphs and Semantics)

Anton Kolonin. Graphs, Metagraphs, Hypergraphs and applications
https://aigents.com/papers/2019/ontologies-language-aigents-2019.pdf
https://aigents.com/papers/2018/hyper-multi-graphs-2018.pdf
https://blog.singularitynet.io/graphs-part-1-how-singularitynet-will-leverage-opencog-aigents-b21e581cf9f8
https://blog.singularitynet.io/graphs-part-2-how-graphs-are-used-in-unsupervised-language-learnings-6a853fc25a29

Second Block (Social Networks and Reputation Systems)

Anton Kolonin et. al. Social Computing and Reputation Systems
https://aigents.com/papers/2019/Social-Computing-with-Online-Communities-2019.pdf
https://aiforgood2019.github.io/papers/IJCAI19-AI4SG_paper_28.pdf
https://www.youtube.com/watch?v=5Pi973JPbZA
https://aigents.com/papers/2018/Aigents-Social-Computing-2018-en.pdf
https://aigents.com/papers/2017/Assessment-of-personal-environments-in-social-networks-2017-IEEE.pdf
https://arxiv.org/pdf/1806.07342.pdf

Segaran Toby. Programming Collective Intelligence
http://shop.oreilly.com/product/9780596529321.do (free and Russian versions exist)

Robert B. Cialdini. Influence: The Psychology of Persuasion
https://www.amazon.com/Influence-Psychology-Persuasion-Robert-Cialdini/dp/006124189X
(free and Russian versions exist)

Anton Kolonin. References
http://aigents.com/en/references.html (free and Russian versions exist)
https://steemit.com/@aigents

https://medium.com/@aigents
https://golos.io/@aigents

Third block (Natural Language Processing)

Anton Kolonin et. al. Unsupervised Language Learning
https://aigents.com/papers/2019/open-cog-atomspace-language-2019.pdf
http://agi-conf.org/2019/wp-content/uploads/2019/07/paper_8.pdf
http://agi-conf.org/2019/wp-content/uploads/2019/08/UnsupervisedLanguageLearningAGI2019.pdf
https://www.youtube.com/watch?v=cwgtcOfA3KI
http://langlearn.singularitynet.io/data/docs/UnsupervisedLanguageLearningAGI2018_text.pdf
http://langlearn.singularitynet.io/data/docs/UnsupervisedLanguageLearningAGI2018.pdf
https://www.youtube.com/watch?v=ABvopAfc3jY
https://blog.singularitynet.io/an-understandable-language-processing-3848f7560271
https://blog.singularitynet.io/bridging-the-language-divide-e2be43f3a37a
https://arxiv.org/pdf/1401.3372.pdf

Anton Kolonin. Approaching Bot Development
https://aigents.com/papers/2017/Bots-and-Aigents-2017-ru.pdf (English version should be available)

Diploma thesis

- Background
    - http://aigents.com/papers/2019/AI-state-2019-en.pdf
        - http://aigents.com/papers/2019/ExplainableLanguageProcessing2019.pdf
        - http://aigents.com/papers/2019/Reputation-Systems-for-Online-Environments-2019.pdf
- Sample themes for diploma thesis
    - Explainable/interpretable natural language processing
        - Grammar learning for <language of your choice>
        - Language generation for <language of your choice> (except English)
        - Sentiment evaluation for <language of your choice> (except English)
        - Sentiment modulation for <language of your choice>
        - Semantic Extension to Link Grammar Parser for domain-specific syntax-to-semantics mapping <language of your choice>
    - Social graph analysis
        - Social structure and dynamics mining for <social/media network of your choice>
            - Selecting the topic
                - Finding and studying opinion leaders in social media
                    - https://medium.com/@aigents/aigents-social-computing-for-personalized-social-connectivity-and-reputation-f4b0ed6905a0
                    - https://www.google.com/search?q=finding%20opinion%20leaders
                    - https://paperswithcode.com/task/leadership-inference
                    - https://arxiv.org/abs/1806.07342
                    - https://steemit.com/ai/@aigents/aigents-finding-opinion-leaders-on-steemit
                - Clustering communities based on social media data (Community detection based on social media data)
                    - https://www.google.com/search?q=community%20detection

        - https://paperswithcode.com/task/community-detection
        - https://en.wikipedia.org/wiki/Formal_concept_analysis
    - Exploring sentiment dynamics in social media (public sources)
- Explore the subject domain, prepare the overview on available techniques/algorithms, write “Introduction/Overview” part of the thesis - 1st semester
- Explore the possible sources of the data - 1st semester
    - Real field data - using one of the social networks/media sources (hard to get but real), referring to existing Aigents social plugins https://github.com/aigents/aigents-java/tree/master/src/main/java/net/webstructor/comm and storing data in unified format, referring to existing implementation
        - Pick one of the sources below (first)
            - LinkedIn,
            - Weibo (API is granted for companies only?)
            - Twitter (Aaron, Zhang Xu)
            - TikTok,
            - QQ,
            - WeChat,
            - Medium,
            - Instagram
            - VKontakte
            - etc.
        - Select the implementation technology (next)
            - Using Java, you can reuse some of the https://github.com/aigents/aigents-java/tree/master/src/main/java/net/webstructor/comm code
            - Using Python, reuse some open source code
            - Using Python, write code of your own from the scratch
    - Public data sets like mentioned in
        - https://paperswithcode.com/task/leadership-inference
        - https://paperswithcode.com/task/community-detection
    - Simulate synthetic data (easy to generate but not real), referring to the existing SingularityNET simulation code https://github.com/singnet/reputation
- Decide where to store the Data - 2nd semester
    - PostgresDB (need to implement the custom schema)
    - Timescale DB (need to implement graphs)
    - Grakn/Neo4J (need to augment with time)
    - Aigents Graphs (need to get used to)
- Evaluate existing algorithms on the following data sets, provide preliminary conclusion and defend the coursework - 2nd semester (coursework in May)
    - Simulated synthetic data
    - Public data sets
- Extending the work, making sense of the results - 3-4 semesters
    - Involve more techniques/algorithms or invent improvements to existing ones if needed
    - Involve more data such as
        - Public data sets
        - Real field data
    - Present the results and summarize the work

- Bonus: Evaluating possible practical extensions to basic results
  - Recommending contributors
  - Recommending sources
  - Building teams
  - Finding typical temporal patterns
  - Finding specific temporal patterns leading to events of interest
- Sentiment analysis in Social graphs for <social/media network of your choice> and <language of your choice>
- Socio-economic modeling for distributed marketplaces (decentralized finance)

- Terms
  - High motivation and self-organization
  - Java or Python skills or ability to acquire them quickly
  - Acceptance based on initial project involving programming

Sample themes for coursework

- Social Media Analysis
  - Identifying opinion leaders by mentions
    - Resources
      - https://github.com/singnet/reputation/issues/280
      - https://arxiv.org/pdf/1806.07342.pdf
    - Data Options
      - Cryptofinance news feed (the link is given)
    - Technology
      - From the scratch and/or any open source findings - flexible :-)
      - https://github.com/singnet/reputation
      - https://github.com/aigents/aigents-java/blob/master/src/main/java/net/webstructor/peer/Reputationer.java
    - Results Destination
      - Any github/gitlab under MIT license, using Java or Python
      - TBD based on progress
    - Subtasks (Tentative)
      - Identify entities, being accounts impersonating people or organizations (build a registry)
        - Feed Owners (about 30)
        - User mentions (???)
      - Count the mentions and build the weighted graph of the mentions across the entities
      - Apply liquid rank algorithm based on mentions and provide the ranking list
        - https://aigents.com/graphs.html
        - https://aigents.com/ui/aigents-graph.js?24
          - See //// Node order computations ////
          - directed : function(links,weighted,linktypes,iterations){
        - https://aigents.com/ui/aigents-al.js?24
      - Have comparison, based on few different ways to rank the account
        - Just amount of mentions across all channels
        - Amount of unique individual mentions across channels (so any number of mentions of user A is channel of user B is counted once)
        - Liquid rank

        - Number of actual followers (ground truth???)
  - Identifying sentiment on a personal basis (Kirill?)
  - Clustering media sources
- Content Analysis (see https://aigents.github.io/inlp/)
  - Relationship Extraction
    - Resources (optional)
      - https://ieeexplore.ieee.org/document/4427566
      - https://www.youtube.com/watch?v=YY7IMVFWTzo
      - https://aigents.com/papers/2021/InterpretableLanguageProcessing2021.pdf
      - https://youtu.be/pjIhBId8zMc
      - http://aigents.com/papers/2015/Kolonin-Automatic-text-classification-and-property-extraction-SIBIRCON-2015-slides.pdf
      - https://aigents.com/papers/2015/Kolonin-Automatic-text-classification-and-property-extraction-SIBIRCON-2015.pdf
      - https://docs.python.org/3/library/re.html
    - Data options
      - Cryptofinance news feed (link will be given) - practical
      - Any - flexible :-)
    - Technology
      - From the scratch and/or any open source findings - flexible :-)
      - Aigents patterns
        - https://github.com/aigents/aigents-java/blob/master/src/main/java/net/webstructor/al/Reader.java#L114
        - https://github.com/aigents/aigents-java/blob/master/php/agent/agent_sites.php#L334
    - Results Destination
      - Any github/gitlab under MIT license, using Java or Python
      - TBD based on progress
    - Subtasks (Tentative)Subtasks (Tentative)
      - Identify key entities (accounts, people, organizations, currencies) that may have relationships expressed in the text
      - Make it possible to tag the entities in the corpus and count the occurences
      - Make it possible to tag the co-occurring entities in the corpus and build the weighted graph of the co-occurrences
      - Make it possible to identify the property (meaning) of the co-occurrence of the entities in the corpus and build the weighted labeled graph of the co-occurrences
  - Trending Topic Detection
  - Ontology building by topic clustering
  - Grammar mining (“The Arrival”)
    - Resources:
      - https://aigents.github.io/inlp/
      - http://langlearn.singularitynet.io/
      - http://aigents.com/papers/2020/InterpretableLanguageProcessing2020.pdf
      - https://www.youtube.com/watch?v=FzKMtNILmDk
    - Data options:
      - Cryptofinance news feed (link will be given) - practical
      - http://langlearn.singularitynet.io/data/poc-english/poc_english.txt - simple
      - https://en.wikipedia.org/wiki/Voynich_manuscript - challenging
      - Any - flexible :-)
    - Technology

            - From the scratch and/or any open source findings - flexible :-)
            - http://langlearn.singularitynet.io/
        - Results destination
            - https://github.com/aigents/aigents-java-nlp - Java
            - https://github.com/singnet/language-learning/ - Python
            - Any github/gitlab under MIT license, using Java or Python
        - Subtasks (Tentative)
            - Create vector space for words based on word embeddings, directional embeddings, etc.
            - Cluster word in vector space searching for categories representing “parts of speech” or “topic-specific terms”
            - Build hierarchical trees of categories searching for a taxonomy
            - Build heterarchical directed graphs of categories searching for an ontology
            - Suggest a way to find relationships between the categories in the tree
- Multi-Agent Interactions and Experiential Learning
    - Simulating Limit Order Book Market Trading
    - Simulating Limit Order Book Market Making
- Graph Visualization
    - Visualizing the structure of features and knowledge learned by ML models (XAI/IAI)
        - Visual objects and scenes
        - Sentiment analysis applications
    - Hypergraph visualization
    - Metagraph visualization
    - Graph visualization for multiple link types with arc connectors
    - Aigents Graphs - porting from JavaScript to Python
    - Aigents Graphs - porting from JavaScript to Java
    - Building ontology for a domain
        - Building and visualizing NLP ontology for <TBD> language
            - Use the language of choice and visualize it using open-source software, referring to ideas in http://aigents.com/papers/2019/ExplainableLanguageProcessing2019.pdf
                - Building and visualizing NLP ontology based on <TBD>
                    - Use NLP formalism and rendering tools of choice
                - Visualizing exiting Link Grammar dictionaries - see example at http://langlearn.singularitynet.io/graph.html (see “View Link Grammar”)
                    - Port http://langlearn.singularitynet.io/graph.html
                        - to Python-based rendering
                        - to Java-based rendering (can refer to Link Grammar dictionary reader https://github.com/aigents/aigents-java-nlp)
                    - Custom visualization
                        - Python-based rendering
                        - Java-based rendering (can refer to Link Grammar dictionary reader https://github.com/aigents/aigents-java-nlp)
                        - JavaScript-based rendering
        - Building and visualising ontology for <TBD> subject domain based on Schema.org model
            - Use subject domain of choice, describe it extending the https://schema.org/ and visualise it using open-source software
- Social Computing

- Social Network plugin (for ~~Facebook, VKontakte~~, ~~Slack, Telegram, Steemit, Golos, Reddit, Twitter,~~ Instagram, LinkedIn, WeChat, TikTok, Medium)
    - Instagram
    - LinkedIn
    - ...
- Bot/Troll behavior detection
- Reputation System
    - Reputation in Marketplaces (Predictiveness, Behavioral Patterns)
        - Based on simulation and reputation system in the code forked from https://github.com/singnet/reputation
            - Run the experiments for different configurations of simulated marketplaces under different scamming behavioral patterns
            - Try to improve the code (add extra function for adjustment of the reputation system parameters) so that it can recognize the change in behavioral patterns of scammers and adjust the parameters of the reputation system accordingly
            - **Bonus:** Contribute the new code function to base repository with unit test
    - Reputation in Communities (Social Interactions) - using the algorithm described under the link, applied to either synthetic data or simulation of the data for “virtual community”.
        - Based on simulation and reputation system in the code forked from https://github.com/singnet/reputation
            - Create new simulation module implementing behavior in a social network or a bunch of messenger group by “conventional participants” and “scammers” or “spammers”
            - Run the experiments for different configurations of simulated community under different parameters of the reputation system
            - Explore what kind of the reputation system settings provide better identification of scammers/spammers.
            - **Bonus:** Contribute the new code function to base repository with unit test
    - Reputation in Blockchains (Resistance to Attacks on Consensus)
        - Refer to
            - https://steemit.com/blockchain/@aigents/proof-of-reputation-as-liquid-democracy-for-blockchain
            - https://blog.singularitynet.io/reputation-consensus-and-liquid-democracy-based-on-open-ledger-protocol-for-distributed-ai-systems-ffc2e6f387d1
            - https://docs.google.com/document/d/1izovbeKmUSm253zfx-y0Dxg6aczkzU_Nnim2-fYmEX4/edit#heading=h.bclhs7fgruhm
            - Anything relevant found on Google
        - Implement simulation and consensus based
            - https://github.com/singnet/reputation
            - Anything relevant open-source found anywhere
            - Custom implementation from the scratch
- Community Structure Study
    - Identification of social patterns in online communities (finding leaders, hubs, groups, circles) - for given ecosystem, using suitable graph analysis techniques
    - Detection of information flows and distribution channels in online communities
    - Measuring society metrics such as entropy https://arxiv.org/abs/1905.09772

- General Intelligence and Neuro-Symbolic integration
    - XAI/IAI - Extracting semantics from neural network
    - Transfer learning - transfer knowledge across neural networks
    - Adaptive Reinforcement Learning
        - Based on either
            - OpenGym https://gym.openai.com/ examples:
                - https://gym.openai.com/envs/Breakout-v0/
            - Virtual Atari Breakout simulation https://github.com/aigents/aigents-java/blob/master/src/main/java/net/webstructor/util/AgiTester.java
                - In Java
                - Ported to Python
        - Using either
            - Neural networks and Reinforcement Learning
                - http://blog.jzhanson.com/blog/rl/project/2018/05/28/breakout.html
            - Non-axiomatic reasoning system (NARS)
                - https://github.com/opennars/OpenNARS-for-Applications
            - Solution in https://github.com/aigents/aigents-java/blob/master/src/main/java/net/webstructor/util/AgiTester.java
                - In Java
                - Ported to Python
- Unsupervised Language Learning and Conversational Intelligence and Interpretable NLP
    - Text statistics analysis (for morphology, dictionary, word links, word groups, patterns)
    - Formal Concept Analysis for grammar space
    - Extracting DNF/CNF Patterns from networks
    - Text pattern matcher
    - Text parser (MST-Parser, dependency parser, etc.)
        - Unsupervised parser learning
            - MST-Parser https://arxiv.org/pdf/cmp-lg/9805009
            - Parsing Challenge: https://github.com/singnet/language-learning/issues/220
            - Unsupervised parsing materials and grammar learning:
                - http://langlearn.singularitynet.io/data/docs/ (read only parsing part)
                - https://www.youtube.com/watch?v=ABvopAfc3jY (view only parsing part)
                - https://www.youtube.com/watch?v=cwgtcOfA3KI (view only parsing part)
    - Sentiment analysis (using m-skip-n-grams and/or morphology)
        - Option 1: Using m-skip-n-grams
            - Refer to existing open-source implementation and model n-gram (phrases) files https://blog.singularitynet.io/aigents-sentiment-detection-personal-and-social-relevant-news-be989d73b381
            - Use the implementation above of fork it ot create similar one of your own using programing language of your choice, relying on the data files above relying on model n-gram (phrases) files
            - Find existing test/train corpora for sentiment analysis online and evaluate your solution baseline metrics (accuracy, F1, recall, precision) against the test corpora
            - Bonus:

        - Option 1: Compare the baseline metrics of your solution with use of alternative open-source NN solution for sentiment analysis trained on the same train corpora and find the winning solution
        - Option 2: Use existing train corpora to learn the model n-gram (phrases) files by means of supervised learning technique of your choice (overriding or extending the original model files) and compare the results for respective test corpora using both new model and original model and compare the results.
    - Option 2: Using morphology
      - TBD
    - Use language of your choice
      - English
      - Russian
      - etc.
  - Language generation/production (with Link Grammar or any other)
  - Pattern-matching-based chat bot (with ChatScript or any other)
  - Neural-network-based chat bot (TBD)

Detailed course plan

- Block 1-1 Week (2 lectures)
  - Coursework
  - Introduction to Social Computing
    - https://aigents.com/papers/2021/Collective-General-Intelligence-2021.pdf
    - http://aigents.com/papers/2020/Global-Brain-and-AGI-2020.pdf
    - https://www.youtube.com/watch?v=flZj4bmGSzg
- Block 1-2 Week (1 lecture)
  - Introduction to Interpretable AI
    - Introduction to AI
    - GDPR and XAI
    - AGI and Cognitive Architectures
    - Probabilistic inference and composite truth values
    - Semantic graphs vs Neural networks (symbolic vs. sub-symbolic approaches to AI)
    - Graphs, hypergraphs, metagraphs
    - OpenCog's AtomSpace architecture
    - Atomese expressions
    - Slides
      - https://aigents.com/papers/2020/Hybrid-Neuro-Symbolic-2020-en.pdf
      - https://aigents.com/papers/2021/Towards-Interpretable-AGI-2021-en.pdf (slides 7, 8, 9, 10, 11)
    - Video:
      - https://www.youtube.com/watch?v=ZusjVvxEinw
      - https://www.youtube.com/watch?v=a-EH3Dh1y2Y
  - Introduction to InterpretableNLP
    - http://aigents.com/papers/2020/InterpretableLanguageProcessing2020.pdf
    - https://www.youtube.com/watch?v=FzKMtNILmDk
    - https://aigents.github.io/inlp/
- Block 1-3 Week 1 (lecture)
  - Graphs

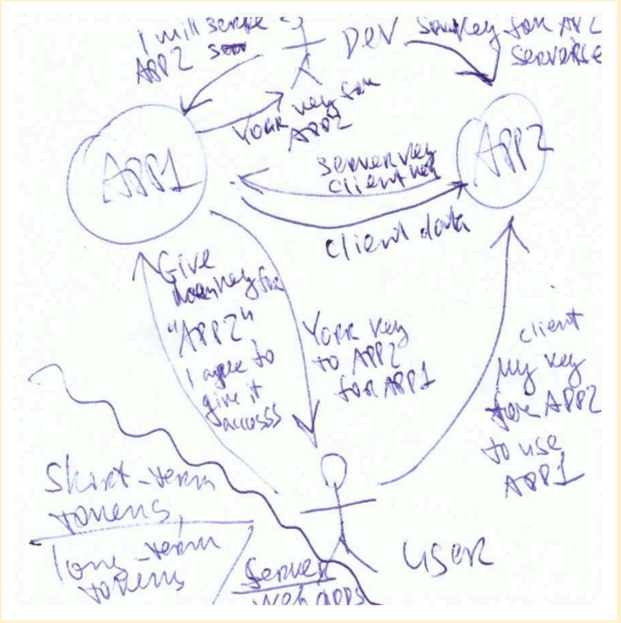
        - https://aigents.com/papers/2018/hyper-multi-graphs-2018.pdf / http://webstructor.net/papers/2018/hyper-multi-graphs-2018.pdf (part 1)
            - Continue on Slides: 51, 52, 18

          https://aigents.com/papers/2021/data4agi.pdf
    - Graph types and visualization issues
        - https://blog.singularitynet.io/graphs-part-1-how-singularitynet-will-leverage-opencog-aigents-b21e581cf9f8
    - Graph querying and examples
        - https://aigents.com/papers/2018/hyper-multi-graphs-2018.pdf / http://webstructor.net/papers/2018/hyper-multi-graphs-2018.pdf (part 2)
        - http://aigents.com/papers/2020/Data-Structures-2020-en.pdf (slide 34)
    - Slides:
        - http://aigents.com/papers/2020/Data-Structures-2020-en.pdf
    - Video
        - https://www.youtube.com/watch?v=-Rk30P2vajo
        - https://www.youtube.com/watch?v=VK6XsDxdqdc
- Block 1-4, Week 3 (lecture)
    - Ontologies, upper ontologies, foundation ontologies and semantic languages http://aigents.com/papers/2019/ontologies-language-aigents-2019.pdf / http://webstructor.net/papers/2019/ontologies-language-aigents-2019.pdf
- Block 1-5, Week 4 (lecture)
    - Foundation/Upper Ontology and Semantic Data http://aigents.com/papers/2019/ontologies-language-aigents-2019.pdf (slide 3)
    - Semantic Web and RDF/OWL history https://aigents.com/papers/2018/hyper-multi-graphs-2018.pdf
        - Standards
            - Data: RDF https://www.w3.org/RDF/ https://www.w3schools.com/xml/xml_rdf.asp
            - Metadata: OWL https://www.w3.org/OWL/ https://www.w3.org/TR/owl-xmlsyntax/apd-example.html
            - Data: JSON/LD https://moz.com/blog/json-ld-for-beginners
                - https://aigents.com/graphs.html
            - Data: Turtle https://www.w3.org/TR/turtle/
        - Ontologies
            - Google Knowledge Graph (Freebase) https://www.google.com/ (search Einstein)
                - http://aigents.com/papers/2019/ontologies-language-aigents-2019.pdf (Slide 4)
            - Schema.org https://schema.org/ (see: https://schema.org/docs/gs.html#microdata_embedded) https://www.w3.org/TR/microdata/
                - https://storyneedle.com/a-visual-approach-to-learning-schema-org-metadata/
            - Cyc https://www.cyc.com/
                - https://www.researchgate.net/figure/A-modified-fraction-of-the-IEEE-SUMO-ontology-21-with-more-than-60-classes-expanded-and_fig8_221466936
            - Wikidata https://www.wikidata.org/ (search Einstein)
                - https://www.wikidata.org/wiki/Q937
            - DBPedia https://wiki.dbpedia.org/
                - http://dbpedia.org/page/Albert_Einstein

            - FOAF (Facebook) http://xmlns.com/foaf/spec/
    - Graph rendering layouts, queries, filtering, building - https://blog.singularitynet.io/graphs-part-3-aigents-graph-analysis-for-blockchains-and-social-networks-142fc8182389
        - Filtering (in local memory, quick)
            - https://aigents.com/graphs.html
        - Building (from remote source)
            - https://aigents.com/
            - https://steemit.com/@themarkymark
- Block 2-1, Week 5 (lecture)
    - Social Evidence and Social Proof, Social Cognitive Model
        - https://www.facebook.com/groups/socialintelligence/permalink/2560675584010502/
        - R.Cialdini https://www.amazon.com/Influence-Psychology-Persuasion-Robert-Cialdini/dp/006124189X
        - https://aigents.com/papers/2015/Kolonin-Computable-cognitive-model-SIBIRCON-2015-slides.pdf
            - http://webstructor.net/papers/2015/Kolonin-Computable-cognitive-model-SIBIRCON-2015-slides.pdf
- Block 2-2, Week 6 (lecture)
    - Social Computing Systems, Blockchains, Distributed Ledger Systems, Peer-2-Peer systems
        - Social computing, Social consensus, Distributed systems, Offline and online
            - http://aigents.com/papers/2019/Social-Computing-with-Online-Communities-2019.pdf
        - Democracy
            - https://aigents.com/papers/2019/Reputation-Systems-for-Online-Environments-2019.pdf (slides 6-7)
        - Byzantine Generals problem - BFT (Byzantine Fault Tolerance)
        - Swarm Intelligence
            - Military
            - Transport
        - Robotics
            - “Society of Mind” M.Minksy
            - “Thousand Brain Theory” (Blind men and an Elephant)
            - Active Perception
        - Distributed Ledger
                - https://101blockchains.com/blockchain-vs-hashgraph-vs-dag-vs-holochain/
            - Blockchain
            - Hashgraph and DAG
            - Consensus protocols
                - https://aigents.com/papers/2019/Reputation-Systems-for-Online-Environments-2019.pdf (slide 9)
                - https://medium.com/the-daily-bit/9-types-of-consensus-mechanisms-that-you-didnt-know-about-49ec365179da
            - Limits
                - Eventual takeover
                - Decentralization instead of distribution
                - Snapshots
- Block 2-2, Week 7 (lecture)
    - Social Computing Systems (continued)

        - Distributed Cryptocurrencies
            - Bitcoin, Ethereum, Steemit, Golos, EOS (POW => DPOS), Gram, Libra
            - Smart Contract Applications: DeFi
        - P2P networks
            - File exchanges
            - Distributed Messengers
                - FireChat
                - Bridgefy
            - Social networks
                - Diaspora
                - Solid from Tim Berners-Lee
        - Collaborative Decision Making (“Liquid Democracy”)
            - Voting
            - ...
    - Reputation/Rating Systems:
        - Social Credit systems
            - https://en.wikipedia.org/wiki/Down_and_Out_in_the_Magic_Kingdom
        - Reputation Management vs. Reputation Systems
            - https://www.youtube.com/watch?v=o0WztEvVyR0
        - https://aigents.com/papers/2019/Reputation-Systems-for-Online-Environments-2019.pdf
            - *http://webstructor.net/papers/2018/Reputation-Systems-for-Liquid-Democracy-2018.pdf*
            - KYC https://en.wikipedia.org/wiki/Know_your_customer
    - Social Media Analytics:
        - https://aigents.com/papers/2017/Personal-Analytics-In-Social-Networks-2017-en.pdf
            - http://webstructor.net/papers/2017/Personal-Analytics-In-Social-Networks-2017-en.pdf
        - https://aigents.com/papers/2017/Aigents-News-Analytics-2017-conf.pdf
            - http://webstructor.net/papers/2017/Aigents-News-Analytics-2017-conf.pdf
        - Content relevance assessment
        - Bot detection
        - Partisanship/bias assessment
- Block 2-2, Week 8 (lecture)
    - Social network applications
        - Unified login
        - Better personalization
        - Better recommendations
        - Trade service for data for re-sale of the data
        - Steal data
    - Social/Web data access
        - API
        - Scalping
            - Public/private
                - Public pages
                - Private authorized pages (captcha cracking)
            - Static/Dynamic
                - Static text/plain
                - Static text/html
                - Static text/html as DOM
                - Dynamic - DOM (Selenium Web Driver, WebKit)

        - Robot-enabled/disabled
            - Robots-enabled
            - Robots-disabled (Selenium Web Driver, bot farms)
        - Access Rate
            - Unrestricted
            - Time-restricted (captcha cracking, bot farms)
    - Social Network regulations, GDPR, scalping rules, robots and crawlers (API-s, headless-es)
- Block 2-3, Week 9 (lecture)
    - Web API-s, Integration with social networks
        - XML
        - JSON
        - REST (POST/GET/PUT) vs. CRUD (Create/Retrieve/Update/Delete)
        - OAuth



    - Database principles
        - SQL/noSQL(Graph/Object) DBs?
        - In-Memory
    - Open source development
        - Licenses
            - OS
                - MIT
                - GNU
            - Proprietary
        - Public repositories
            - Github/Bitbucket
            - Branches and Forks
- Block 3, Weeks 9 (lecture)
    - Unsupervised Language Learning: http://langlearn.singularitynet.io/data/docs/UnsupervisedLanguageLearningAGI2018.pdf
    - NLP basics
        - Vector spaces
            - He is Dad
            - She is Mom
            - He is Tom
            - She is Mary
        - N-gram/Skip-grams

    - NLP graphs https://blog.singularitynet.io/graphs-part-2-how-graphs-are-used-in-unsupervised-language-learnings-6a853fc25a29
    - Sentiment analysis
        - Irony/Sarcasm detection
    - Chat-bot development: https://aigents.com/papers/2017/Bots-and-Aigents-2017-ru.pdf
        - http://www.webstructor.net/papers/2017/Bots-and-Aigents-2017-ru.pdf
- Block 3, Week 10 (lecture)
    - Text classification, entity extraction and attribution: https://aigents.com/papers/2015/Kolonin-Automatic-text-classification-and-property-extraction-SIBIRCON-2015-slides.pdf
        - http://www.webstructor.net/papers/2015/Kolonin-Automatic-text-classification-and-property-extraction-SIBIRCON-2015-slides.pdf
    - Clustering vs Classification problems (NN's, feature engineering, PCA/FA/FCA)
        - Slides by Dmitry Ilvovsky

Contact information

Anton Kolonin
akolonin@gmail.com
http://webstructor.net/test/courses.html
http://aigents.com/en/references.html
https://medium.com/@aigents
https://www.youtube.com/aigents
https://www.facebook.com/akolonin
https://vk.com/akolonin

Appendices

DNF and CNF

Basics

(x OR y) AND (p OR q OR t) AND (s OR p OR w) - conjunction of disjunctions - CNF

(a AND b) OR (c AND d AND x) OR (k AND l AND m) - disjunctions of conjunctions - DNF

Using for structure learning

Parts of body layers:

(a1 AND b1) OR (c1 AND d1 AND x1) OR (k1 AND l1 AND m1) - disjunctions of conjunctions - DNF

Actual beings layers:

(a2 AND b2) OR (c2 AND d2 AND x2) OR (k2 AND l2 AND m2) - disjunctions of conjunctions - DNF

Interactions between the beings:

(a3 AND b3) OR (c3 AND d3 AND x3) OR (k3 AND l3 AND m3) - disjunctions of conjunctions - DNF