COMPREHENSIVE REVIEW READING LIST for Fall 2020 and Spring 2021
Artificial Intelligence/Machine Learning (AI/ML)

Note by: Zahid Hasan.

1 McCarthy, J., and Hayes, P.(1969) Some Philosophical Problems from the Standpoint of Artificial Intelligence. Machine Intelligence. Vol. 4, p.463-502.

- Review paper
  - 1. What is knowledge, causality
  - 2. Formalism which can be proved about correctness and
  - 3. Open question about (2)
  - connection between philosophical logics and AI
- Problem representations in context of computation!!
- Knowledge and representation (model of world/problem)! - should able to answer question.
- Intelligence definition
  - Epistemological – Representation (This paper)
  - heuristic – Rule based
- Philosophical Questions
  - Necessity of philosophy!!
  - Reasoning program (strategy to solve problem)
  - Representations of the world
  - Notion of 'Can'
- Formalism
  - situation
  - Fluents (function of situation)
  - causality
  - Actions
  - Strategies
  - knowledge and ability
- Open Problems
  - Approximate of results
  - Meanings of Can
  - Frame problem
  - formal literature
  - Probabilities
  - parallel processing
- Discussion of Literature
  - Modal logic
  - Logic of Knowledge
  - tense logics
  - Logic and theory of action
  - Counterfactuals
  - Communication process

2 Tversky, A. and D. Kahneman (1974). Judgment under Uncertainty: Heuristics and Biases. Science 185: 1124-1131.

- How probability is calculated and what causes error (different heuristics)?
- Three heuristics are used to calculate the probability (trial on human survey!!)
  - Representativeness
    - employed when people are asked to judge the probability that an object or event A belongs to class or process B
    - One event to another event probability
    - Insensitivity to prior of outcomes
    - Insensitivity to sample size
    - Misconception of chance
    - Insensitivity to predictability
    - The illusion of validity
    - Misconception of regression
  - Availability
    - employed when people are asked to assess the frequency of a class or the plausibility of a particular development
    - Frequency based
    - Biases due to the retrievability
    - Biases due to the effectiveness
    - Biases of imaginability
    - Illusory correction
  - Adjustment and Anchoring :
    - employed in numerical prediction when a relevant value is available.
    - Insufficient Adjustment
    - Anchoring in the assessment of subjective probability distributions
- Discussion
  - Cognitive biases
- Summary(Only required if needed)

3 Jacob Devlin, et al.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2018.

- Language representation model (Bidirectional **encoder** representations from transformers) – Figure 1 [overall system pre-train and fine tune]
    - *self supervised learning steps*
        - Pretrained with unlabeled data (Fine tune with one additional layers for other down streaming language tasks) [Learns the relationship between sentences and word in sentences – Contextual representation]
        - Word2Vec – Non-contextual embedding
            - Next sentence prediction [text pair reps] (just there!)
            - Masked language model (my kind of MLM :))
    - *supervised learning*
        - Downstream tasks 11
- *Key contribution*
    - BERT pretraining improve scalabilty for the downstream task
    - MLM, NSP
    - State-of-the-Art for 11 tasks
    - Bidirectional training and fine-tuning methods
- Required concepts query, key and values
    - Transformers
    - Encoder / decoder
        - input/output
        - query*key' will make the connection with past present and future
        - Attention mechanism – look forwards and backwards
    - Encoder
        - sees everything at a time!
        - Solves my MLM
    - Decoder
        - Sees one after another.
- firstly understand this {https://jalammar.github.io/illustrated-transformer/} everything is straight forward now, just think of query*key' → 512*512 → Softmax matrix multiplied with the value in each encoder steps
- Uses Transformer Encoder
    - input → [cls], words, 512 words
    - output
        - 768 hidden size
        - 512 embedding
            - Interconnected to each others
            - Can see each other in hidden layer by query-key
        - for each position of input
    - Can be thought of CNN in image
    - bidirectional
        - Masking is required
- A new age of embedding → Empirically powerful!
- Related works (with a bit of Tx learning)
    - Unsupervised feature-based

    - ELMo and Embeddings features'
      - Contextual embeddings
      - language modeling: predict the next words → self-supervised
      - LSTM models
    - ULMFiT
      - Transfer learning
      - Downstream tasks
    - transformer → Encoder-decoder
    - Solves the unidirectional approaches in pretraining! (MLM)
      - GPT left to right. (Sometimes may worse!)
  - Unsupervised fine-tuning based
    - Radford et. al. (GPT), ELMo (Peters et al.)
    - Unlabeled pre-train and supervised down-stream train
    - OpenAI transformer
      - training transformer decoder for LM
      - predicting next words
      - masking future token
      - self-encoder and masking
      - Book data
      - unidirectional
- *Methodology*
  - Model architecture (backbone Vaswani et al transformers) [hidden layers, self attention heads]
  - Multilayer bidirectional transformer
    - BERT base (GPT comparable)
      - transformer block = 12, hidden layer size = 768, and self-attention head A=12
      - 110M params
    - BERT Large
  - Input/Output reps [figure 2]
  - Pre-Training (two tasks) – BoodsCorpus, Wikipedia data
    - Masked LM (3.1) – fill in the blanks
      - 15% of the words
        - 80% of them are masked and 10% replaced with random words and 10% with the original words.
        - Else may learn non-contextual embedding
        - https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270
    - NSP – Relation between sentences
      - Two sentence tasks [cls] sentence [sep]
  - Fine tuning (single text and text pairs)
    - Self-attention layer allows so – bidirectional information
    - Swapping appropriate (task-specific) input/output
    - Paraphase, entailment, VQA, text classification/tagging
    - Sequence classification
    - all parameters are fine-tuned
- Experiments: Task specific models

    - 11 NLP tasks
    - GLUE (https://openreview.net/pdf?id=rJ4km2R5t70)
    - SquAD
    - SWAG
- Performs feature extraction
- Ablation studies
- [More](http://jalammar.github.io/illustrated-bert/)
- Some issues
    - Slow convergence than ELMo
    - Can be optimized for more data
    - Importance of NSP is not well understood
    - Hyper-parameters of masking

4 Fahlman, S. and Hinton,G. (1987) Connectionist Architectures for Artificial Intelligence, IEEE Computer 20(1):100-109.

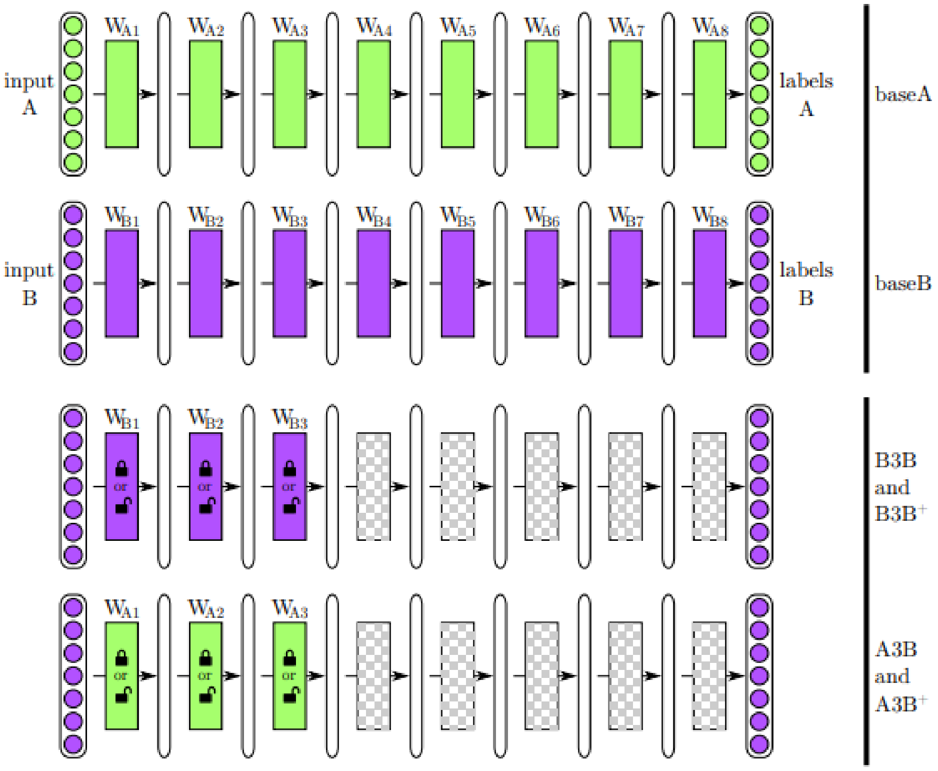
- problem with serial computing
    - Memory (human commonsense) is hard to encode
- Inference in the noisy environment
- representation like human (how)
- requires parallel processing
    - connectionist method?
    - Store pattern in the connections [need many connections]
- connectionism
        - study of a certain class of massively parallel architectures for AI
    - Information stored inside the units
    - Parallelism and search through units
    - Massage passing
- Distributed representation
        - Things in a massively parallel network is to use local representations
- Learning representation
    - Backpropagation
- Hopfield and boltzmann network for constraint satisfaction
    - related to energy function
- Boltzmann machines
        - still successful
    - Pairwise connection only!
    - Problem of gradient descent
- [there are more]

5 Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? In Advances in neural information processing systems (pp. 3320-3328).

- Empirical/experimental analysis paper

- General (starting bottom layers) to specific features (end side top layers) – can these be transferred [studied this transition, experimentally quantify transfer-ability of layers??]
    - initial layers (Gabor filters) – not task specific but *generalized*
    - studied the transition
    - analyze generality vs specificity in each layers
- **Co-apaptive neurons: not wanted**
    - neurons that works together towards particular tasks
    - We want them to learn independently
    - hidden units in a neural networks have highly correlated behavior
    - Dropout reduces co-apaptation
- negative impact on the transfer-ability
    - Higher layers are too specific [for the *task-specific*] – as it should
    - Optimization problem while splitting co-apaptive neurons (!!)
        - kind of unexpected results the received
        - splitting the base network between co-adapted neurons on neighboring layers.
- Image net experiments
    - Who of the previous two dominates, where and how dominates?
        - Tx decrease when distance increases
        - transferring bottom, top or middle layers
    - Basetask and target tasks matters too → How much different are they?
    - Almost better then random initialization (boost the task with fine tuning)
        - Transferring any layers (other is randomly initialize) & fine-tuning is better for generalize → better representation

- General neurons [bottom layers learns filtering], Specific layers [final softmax layers maps task with the neurons]
- research Question (RQ) –
    - Quantify transfer, where transfer occurs [distributed or sharp?], how transition occurs
    - exact nature and extend of transfer-ability?
- Transfer learning
    - Base task [train first] to specific task {source domain}
    - target task – smaller labeled dataset {target domain}
- Contributions
    - Quantify transfer-ability, layer-by-layer transfer-ability and transition.
    - Performance degradation [two issues – see above]
        - How transfer benefit decreases
        - random weights, fine tuning, layer-freezing [in lower section]
    - Layer initialization from base tasks performs better
    - Reports that Transferring from any no of layer works
    - Experiments
        - Overlap and non-overlapping class (A & B)

        - *Selffer* networks (B to B)
        - *Transfer* network (A to B)
        - Use different trained layer for new tasks
        - Figure 1: Example (+ means no freezing)
        - 
input A
W_{A1} W_{A2} W_{A3} W_{A4} W_{A5} W_{A6} W_{A7} W_{A8}
labels A
baseA
input B
W_{B1} W_{B2} W_{B3} W_{B4} W_{B5} W_{B6} W_{B7} W_{B8}
labels B
baseB
W_{B1} W_{B2} W_{B3}
or
B3B and $B3B^+$
W_{A1} W_{A2} W_{A3}
or
A3B and $A3B^+$
    - Experimental setup
        - Similar dataset
        - Dissimilar dataset
        - Random initialization
- Result and discussion (really important reports)
    - Best performing A2B+ (transfer network) → consistently
    - performance drop due to Fragile co-adaptation
        - Fine tune recovers co-adaptive interacion
    - performance drop due to Representation Specificity
        - Negative transfer, very task-specific neurons are not getting retrained

6 Alavi, M. and D. E. Leidner (2001). Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. MIS Quarterly 25(1): 107-136.

- knowledge:
    - abstract notion
    - organizational resource
        - so need KM
- IS research → knowledge management system (KMS)

  - creation, transfer, and application
- This paper: Review and interpretation of knowledge management
- should preserve and build on existing literature.
  - Process view of organizational KM
  - research issues surrounding KM and IT support
- Intro:
  - based on knowledge based prospective
  - In the organizational setup
  - advanced technologies: internet, software agents
  - coding, storing and transmitting knowledge in organization
- Overview of concepts
    - manifold view of knowledge : strategic management
  - Hierarchical view of
    - Distinguish data, information, and knowledge
      - Definition of knowledge!!
  - Alternative perspective on knowledge
    - 1. a state of mind, 2. an object, 3. a process, 4. a condition on information access 5. capability
  - summary of knowledge perspective
    - diff among data, information, and knowledge
    - interpretable by the receiver
    - Hoards of information are of little value: need active processing
  - Taxonomies of knowledge
    - tacit : action, experiences and involvement; cognitive and technical elements
    - explicit: articulate, codified and communication

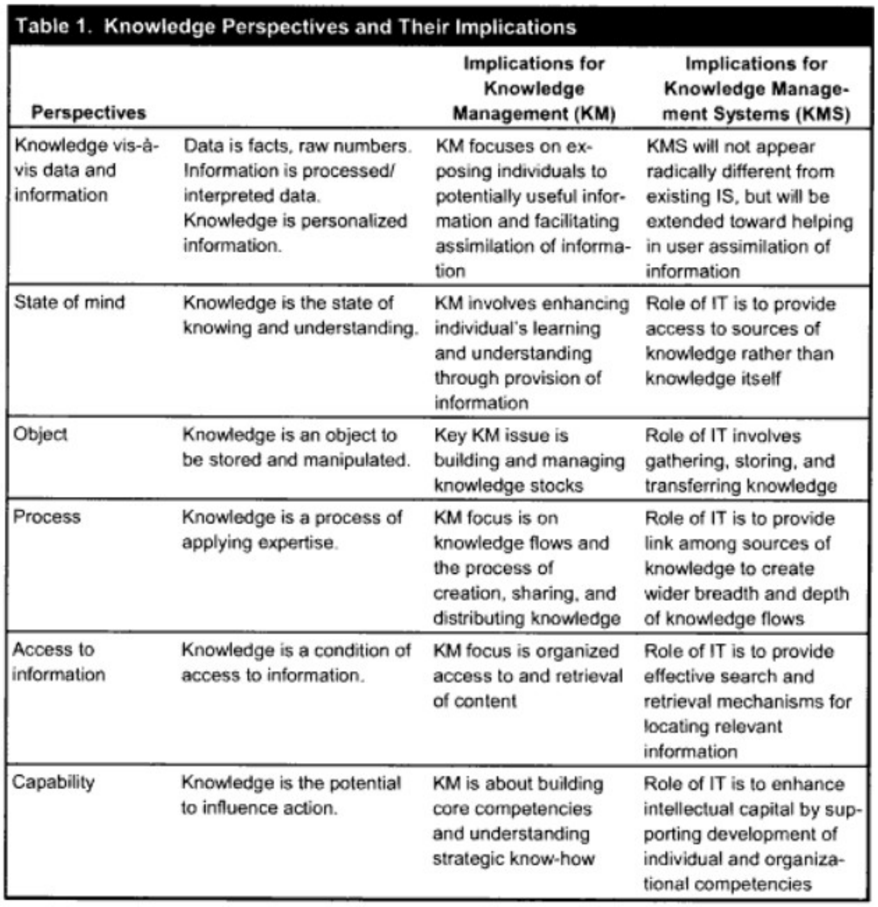


Table 1. Knowledge Perspectives and Their Implications

Perspectives		Implications for Knowledge Management (KM)	Implications for Knowledge Management Systems (KMS)
Knowledge vis-à-vis data and information	Data is facts, raw numbers. Information is processed/ interpreted data. Knowledge is personalized information.	KM focuses on exposing individuals to potentially useful information and facilitating assimilation of information	KMS will not appear radically different from existing IS, but will be extended toward helping in user assimilation of information
State of mind	Knowledge is the state of knowing and understanding.	KM involves enhancing individual's learning and understanding through provision of information	Role of IT is to provide access to sources of knowledge rather than knowledge itself
Object	Knowledge is an object to be stored and manipulated.	Key KM issue is building and managing knowledge stocks	Role of IT involves gathering, storing, and transferring knowledge
Process	Knowledge is a process of applying expertise.	KM focus is on knowledge flows and the process of creation, sharing, and distributing knowledge	Role of IT is to provide link among sources of knowledge to create wider breadth and depth of knowledge flows
Access to information	Knowledge is a condition of access to information.	KM focus is organized access to and retrieval of content	Role of IT is to provide effective search and retrieval mechanisms for locating relevant information
Capability	Knowledge is the potential to influence action.	KM is about building core competencies and understanding strategic know-how	Role of IT is to enhance intellectual capital by supporting development of individual and organizational competencies

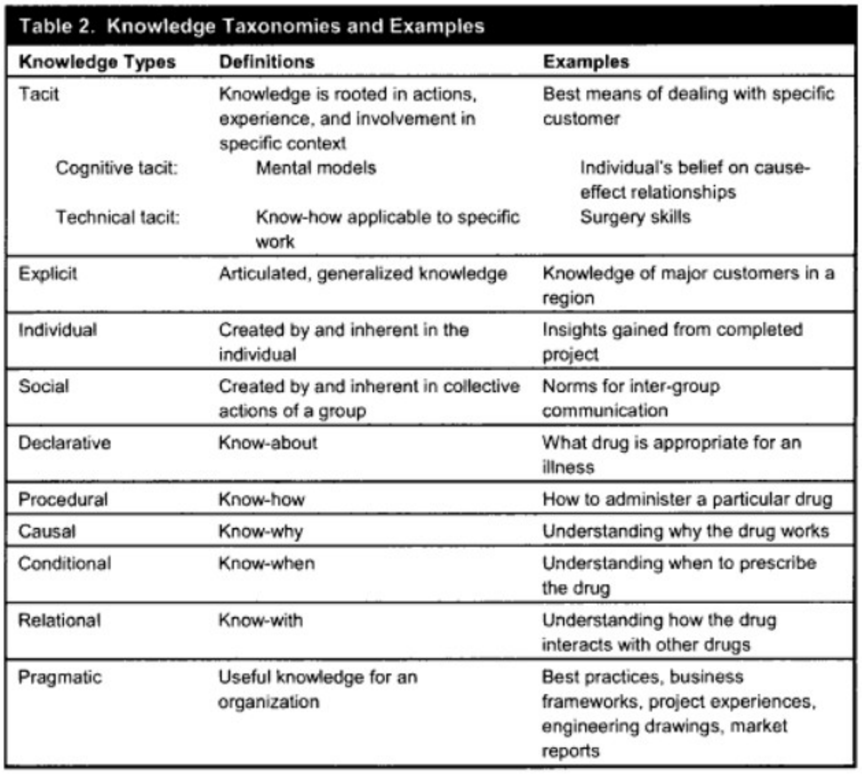
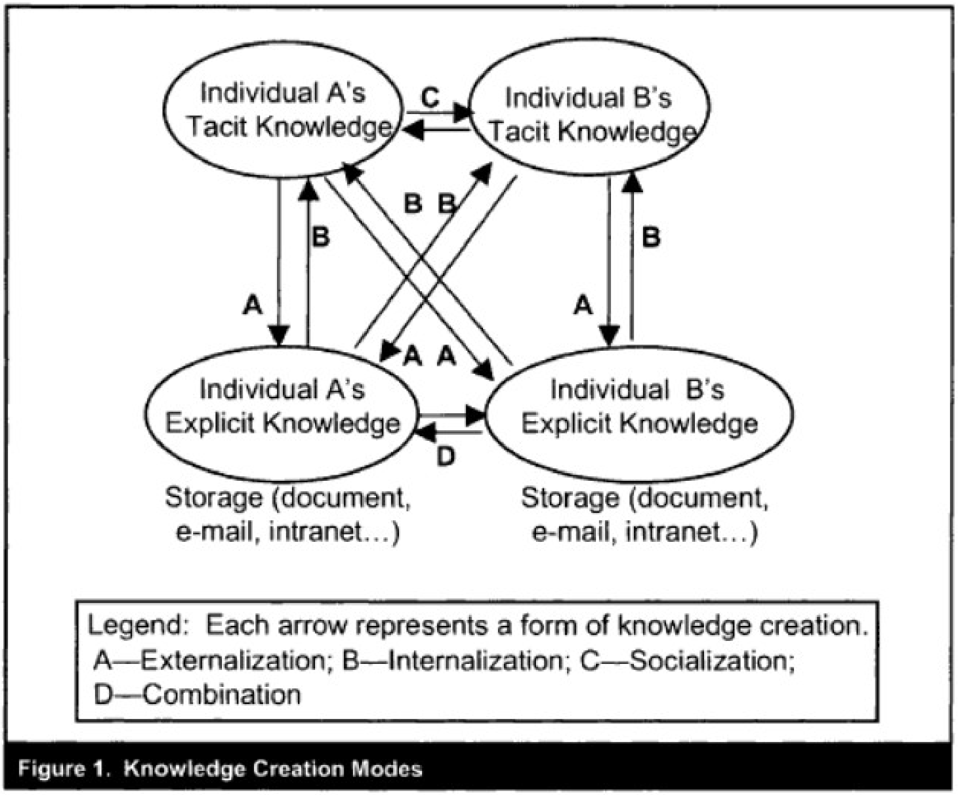


Table 2. Knowledge Taxonomies and Examples

Knowledge Types	Definitions	Examples
Tacit	Knowledge is rooted in actions, experience, and involvement in specific context	Best means of dealing with specific customer
Cognitive tacit:	Mental models	Individual's belief on cause-effect relationships
Technical tacit:	Know-how applicable to specific work	Surgery skills
Explicit	Articulated, generalized knowledge	Knowledge of major customers in a region
Individual	Created by and inherent in the individual	Insights gained from completed project
Social	Created by and inherent in collective actions of a group	Norms for inter-group communication
Declarative	Know-about	What drug is appropriate for an illness
Procedural	Know-how	How to administer a particular drug
Causal	Know-why	Understanding why the drug works
Conditional	Know-when	Understanding when to prescribe the drug
Relational	Know-with	Understanding how the drug interacts with other drugs
Pragmatic	Useful knowledge for an organization	Best practices, business frameworks, project experiences, engineering drawings, market reports

    - KM in organization
        - identify and leverage the collective knowledge in organization to help compete.
    - KMS
        - a class of information system applied to managing organizational knowledge.
- Organizational kM process: a framework from IS
        - creation
        - storage
        - transfer
        - application
    - creation
        - Developing new content or replacing existing contents
        - 4 creation modes
        - 
Individual A's Tacit Knowledge
C
Individual B's Tacit Knowledge
B
B B
B
A
A
A A
Individual A's Explicit Knowledge
Individual B's Explicit Knowledge
D
Storage (document, e-mail, intranet...)
Storage (document, e-mail, intranet...)

Legend: Each arrow represents a form of knowledge creation. A—Externalization; B—Internalization; C—Socialization; D—Combination

Figure 1. Knowledge Creation Modes

    - storage / retrieval
        - semantic or episodic
        - memory
    - knowledge transfer

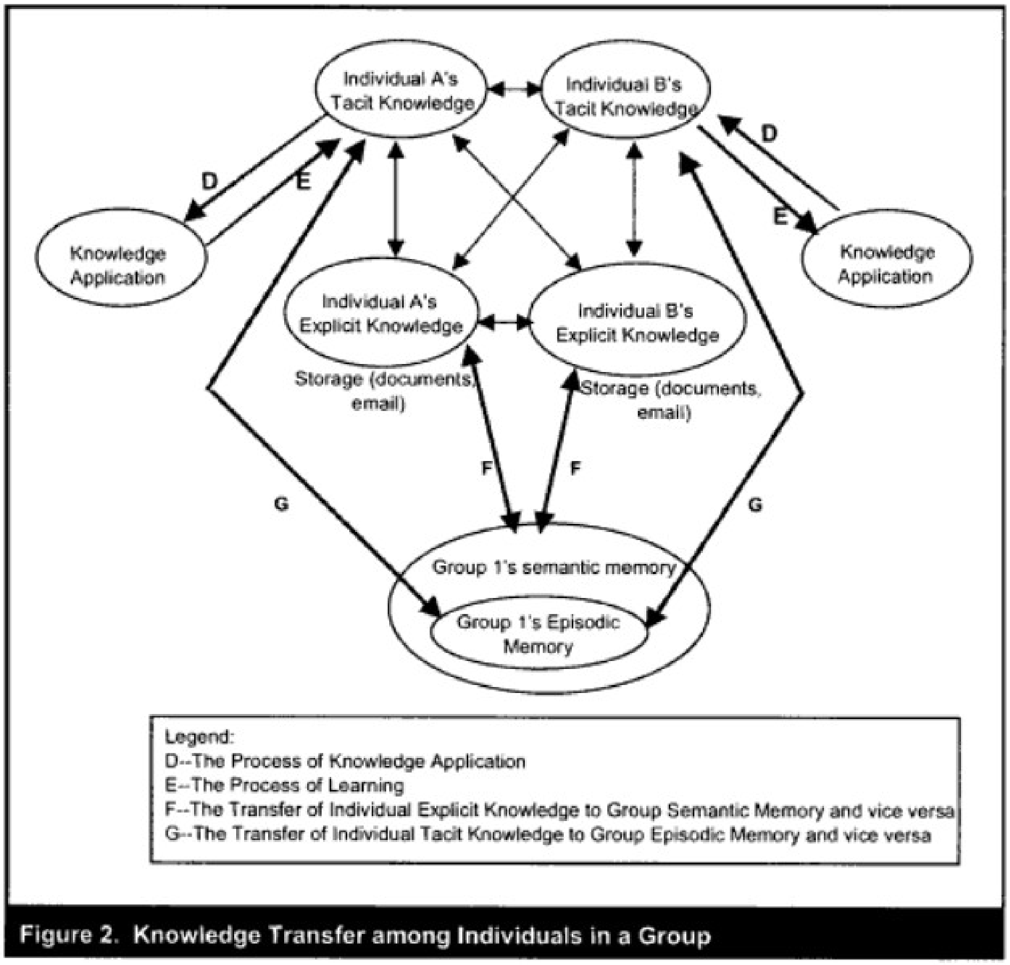

Individual A's Tacit Knowledge
Individual B's Tacit Knowledge
D
E
D
E
Knowledge Application
Knowledge Application
Individual A's Explicit Knowledge
Individual B's Explicit Knowledge
Storage (documents, email)
Storage (documents, email)
F
F
G
G
Group 1's semantic memory
Group 1's Episodic Memory
Legend:
D--The Process of Knowledge Application
E--The Process of Learning
F--The Transfer of Individual Explicit Knowledge to Group Semantic Memory and vice versa
G--The Transfer of Individual Tacit Knowledge to Group Episodic Memory and vice versa

Figure 2. Knowledge Transfer among Individuals in a Group

    -
  - knowledge application
    - knowledge-based theory

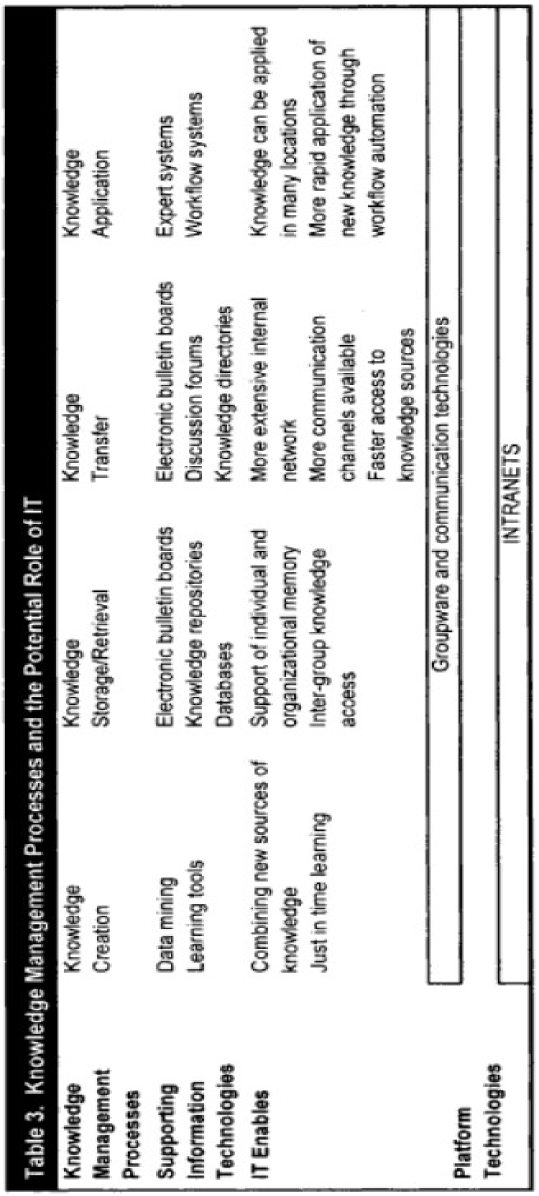
- 

Table 3. Knowledge Management Processes and the Potential Role of IT

Knowledge Management Processes	Knowledge Creation	Knowledge Storage/Retrieval	Knowledge Transfer	Knowledge Application
Supporting Information Technologies	Data mining Learning tools	Electronic bulletin boards Knowledge repositories Databases	Electronic bulletin boards Discussion forums Knowledge directories	Expert systems Workflow systems
IT Enables	Combining new sources of knowledge Just in time learning	Support of individual and organizational memory Inter-group knowledge access	More extensive internal network More communication channels available Faster access to knowledge sources	Knowledge can be applied in many locations More rapid application of new knowledge through workflow automation
Platform Technologies	Groupware and communication technologies			
	INTRANETS			

- Research issue in KM
  - Research issue in creation

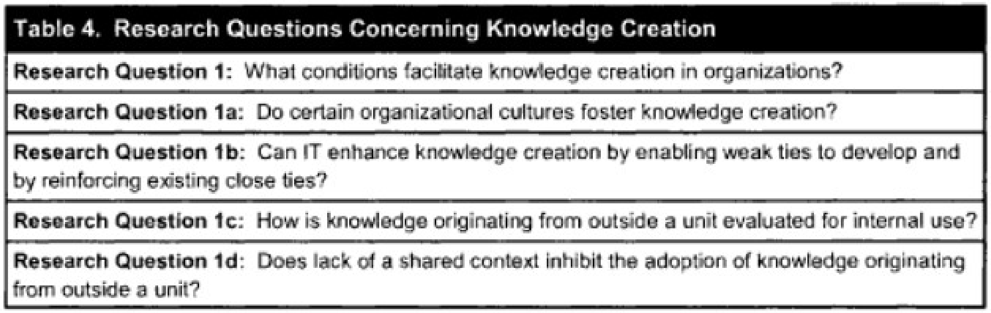


Table 4. Research Questions Concerning Knowledge Creation
Research Question 1: What conditions facilitate knowledge creation in organizations?
Research Question 1a: Do certain organizational cultures foster knowledge creation?
Research Question 1b: Can IT enhance knowledge creation by enabling weak ties to develop and by reinforcing existing close ties?
Research Question 1c: How is knowledge originating from outside a unit evaluated for internal use?
Research Question 1d: Does lack of a shared context inhibit the adoption of knowledge originating from outside a unit?

  - Research issue in store

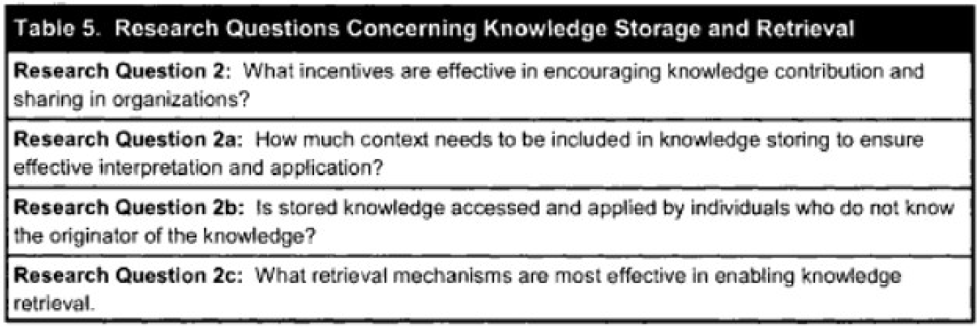


Table 5. Research Questions Concerning Knowledge Storage and Retrieval
Research Question 2: What incentives are effective in encouraging knowledge contribution and sharing in organizations?
Research Question 2a: How much context needs to be included in knowledge storing to ensure effective interpretation and application?
Research Question 2b: Is stored knowledge accessed and applied by individuals who do not know the originator of the knowledge?
Research Question 2c: What retrieval mechanisms are most effective in enabling knowledge retrieval.

- Research issue in transfer

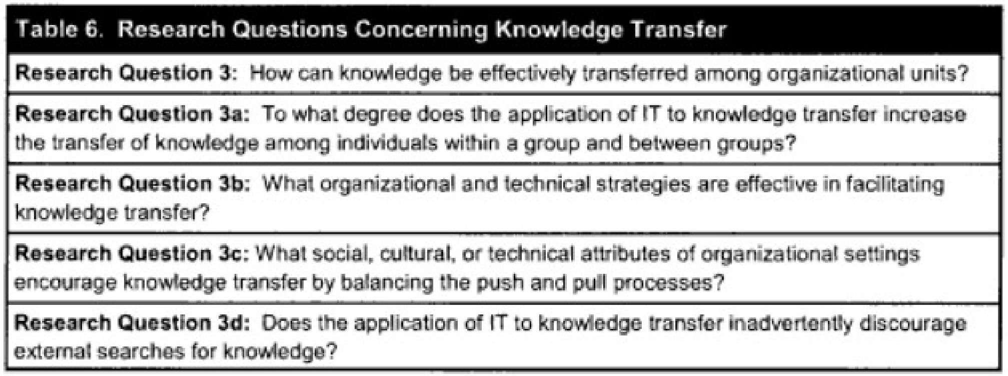


Table 6. Research Questions Concerning Knowledge Transfer
Research Question 3: How can knowledge be effectively transferred among organizational units?
Research Question 3a: To what degree does the application of IT to knowledge transfer increase the transfer of knowledge among individuals within a group and between groups?
Research Question 3b: What organizational and technical strategies are effective in facilitating knowledge transfer?
Research Question 3c: What social, cultural, or technical attributes of organizational settings encourage knowledge transfer by balancing the push and pull processes?
Research Question 3d: Does the application of IT to knowledge transfer inadvertently discourage external searches for knowledge?

- research issue in Knowledge application

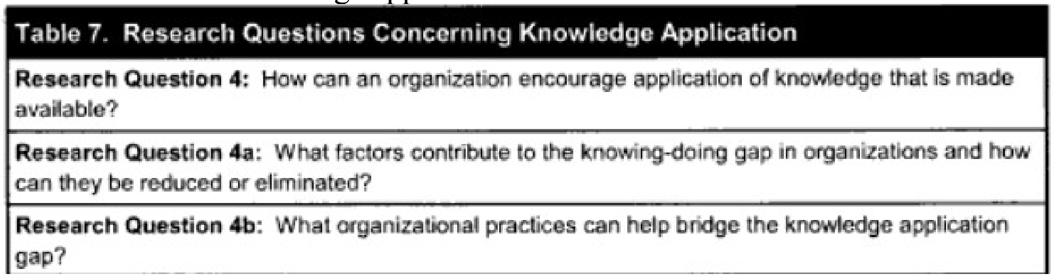


Table 7. Research Questions Concerning Knowledge Application
Research Question 4: How can an organization encourage application of knowledge that is made available?
Research Question 4a: What factors contribute to the knowing-doing gap in organizations and how can they be reduced or eliminated?
Research Question 4b: What organizational practices can help bridge the knowledge application gap?

- IT and KM research issue

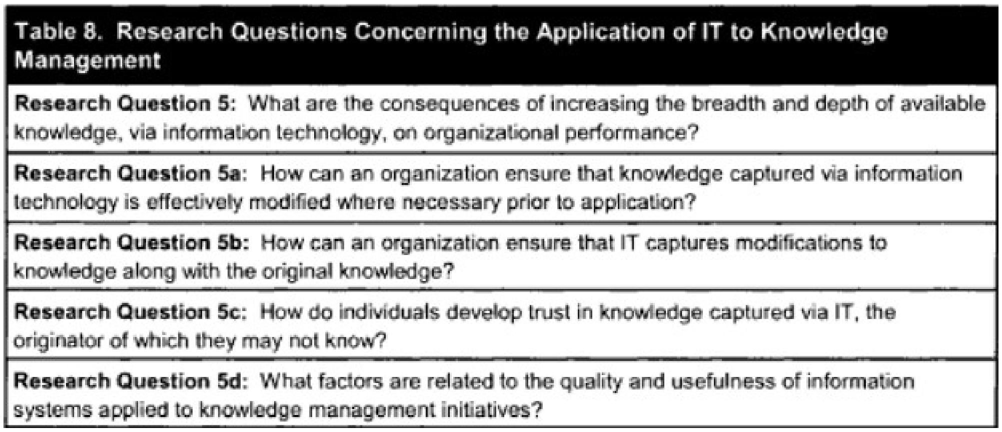


Table 8. Research Questions Concerning the Application of IT to Knowledge Management
Research Question 5: What are the consequences of increasing the breadth and depth of available knowledge, via information technology, on organizational performance?
Research Question 5a: How can an organization ensure that knowledge captured via information technology is effectively modified where necessary prior to application?
Research Question 5b: How can an organization ensure that IT captures modifications to knowledge along with the original knowledge?
Research Question 5c: How do individuals develop trust in knowledge captured via IT, the originator of which they may not know?
Research Question 5d: What factors are related to the quality and usefulness of information systems applied to knowledge management initiatives?

- conclusion (4)
  - reviewed complex nature of organizational KM
  - independent process of Knowledge creation, storage, retrieval, and application
  - draw various IT tools
  - future research direction

7 Bolukbasi, Tolga, et al. "Man is to computer programmer as woman is to homemaker? Debiasing word embeddings." Advances in Neural Information Processing Systems. 2016.

- Summary
  - Problems: ML algorithm can amplify biases of data
  - Observations and shows
    - Geometrically, gender bias is captured by a direction in the word embedding
    - gender neutral words are linearly separable from gender words in embedding
  - Proposes methodologies
    - Modified embedding to remove gender stereotypes
      - queen should be related to female
      - receptionist should not be associated with female
    - Empirical demonstration of methods' success
      - Crowd-worker
      - Standard metrics
- Bias in word embedding (stereotype) – google news data training
  - Quantitatively demonstrate that word-embeddings contain biases in their geometry
  - Geometric direction of vectors to quantify bias (distance metrics)
  - Task (remove bias but keep association) – Interesting
  - Metric to measure bias! (compatible with man-woman) [he/she occupants] by crowd
    - Direct – Clear connection
      - direct bias is measured as the association between a gender neutral word and a gendered word pair: defined in this paper
    - Indirect – Gender neutral words
      - word receptionist is closer to the word softball than it is to the word football
- Related works and preliminaries
  - Gender bias and stereotype in English
  - Bias in algorithm
    - predict repeat offender exhibit indirect racial biases (indirect bias)
  - Word Embedding
  - Crowd experiments → two experiments on AMT
- Geometry of gender bias in word embedding
  - occupational stereotype
  - Analogies exhibiting stereotype
  - Focus only on gender bias
  - I*dentifying gender subspace*
    - ten gender pair difference vectors and computed its principal components (PCs).
  - Direct Bias (defining equation)
  - Indirect Bias (defining equation)
- *Hypothesis:* existence of low dimensional gender bias space (make this projection = 0) in the embedding process
- Algorithm to debias!
  - two tasks
    - Reduce bias & Maintain embedding utility
  - Gender stereotype
    - occupational stereo

        - Analogy with the subtraction between male and female
    - Analogy and stereotype and bias
    - methodologies
        - Identify the gender subspace??
            - take 10 gender pair difference and compute Principal components top one **g**)
        - Debias 1: Hard debiasing
            - Neutralize (forces zero in gender subspace)
            - Equalize/ soften (forces Equidistant with gender words)
                - Observation 1
        - Debias 2: Soft bias correction
            - Optimization problem
- Determining gender neutral words
- Methods
    - Gender specific words to learn gender space
    - embedding consist of sufficient information to reduce the bias.
- *Key contributions*
    - understanding bias in words
    - reduce direct and indirect gender bias
    - bias in society!!
- *Look for the arxiv paper of 25 pages*
- [further read](http://cs229.stanford.edu/proj2016/report/BadieChakrabortyRudder-ReducingGenderBiasInWordEmbeddings-report.pdf)

8 Woodridge, M., & Jennings, N. R. (1995). Intelligent Agents: Theory and Practice. *The Knowledge Engineering Review*,10(2), 115-152.

- Agent
    - theoretical and practical issues in construction of intelligent agent
- 3 areas discussed
    - Agent theory: what an agent is, and the use of mathematical formalismsfor representing and reasoning about the properties of agents
    - agent architecture: as software engineering models of agents
    - agent language: software systems for programming and experimenting with agents;
- Notion of agent
    - two general usages of the term ‘agent’: the first is weak, and relatively uncontentious; the second is stronger, and potentially more contentious.
    - strong and weak notion
- Agent theory
    - intentional system
    - Representing Intentional Notion
    - Possible Worlds Semantics
    - Alternatives to the Possible Worlds Model
    - Pro-attitudes: Goals and Desires
    - Theories of Agency
    - Communication
    - Discussion
    - Further Reading

- Agent Architectures
    - Classical Approaches: Deliberative Architectures
    - Alternative Approaches: Reactive Architectures
    - Hybrid Architectures
    - Discussion
    - Further Reading
- Agent language
    - Discussion
    - Further Reading
- Application
    - Cooperative Problem Solving and Distributed AI
    - Interface Agents
    - Information Agents and Cooperative Information Systems
    - Believable Agents
- what an agent is, how the notion of an agent can be formalised, how appropriate agent architectures can be designed and implemented, how agents can be programmed, and the types of applications for which agent-based solutions have been proposed

9 Lawrence Page, Sergey Brin, Rajeev Motwani, Terry Winograd (1998). The PageRank citation ranking: Bringing order to the Web. 1998. Available at http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf

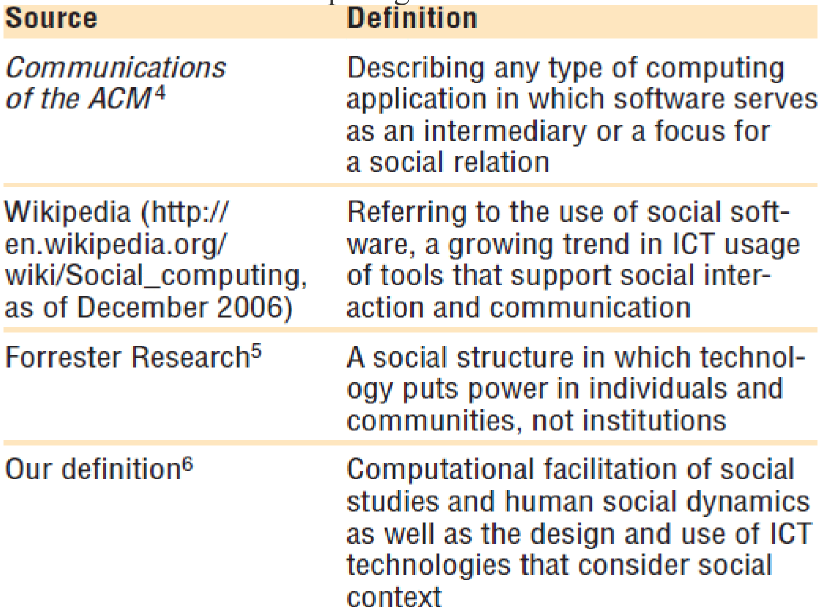
- Importance of Webpage
    - subjective
    - However, there exists relative importance
    - *measure* human interest and devotion to webpages
- Information retrieval from WWW
    - diverse and large
    - cover both experienced and inexperienced users
- hypothesis and goal
    - Hyp: WWW is hypertext and provides considerable auxiliary information on top of the text of the web pages, such as link structure and link text.
    - Goal: aims for creating global importance of webpage
        - by taking advantage of the link structure of the Web
- This paper: PageRank methods
        - computing a ranking for every web page based on the graph of the web
    - Compared with random surfer
    - ORIGIN of GOOGLE
    - Utilizes the hyperlinks in the webpage
    - Application: search, browsing and traffic estimation.
- Artificial increasing the citation count
- Approximation of overall relative importance of the webpages
- Related works
    - Link structure usage

        - Importance of who is citing!
    - PageRank
        - Loop problem (a →b→a)
- usage of link structure of web-pages
    - Graph computation
    - Adjacency matrix
- Propagation of ranking through links
    - A page has high rank if the citing pages has also high link (sum of the ranks)
- PageRank
    - Definition → From simplified to consider ranksource
    - RankSink (no outgoing link, a trap, or loop)
        - issue: Acquire ranks are never distributes
    - Solved by ranksource → New matrix to find the eigenvector
- Random Surfer Model
    - E user defined parameters – Customized page ranks
    - Jumps to random page sometimes if it gets bored (stuck somewhere)
- Computing pagerank
    - power iteration algorithm
        - Dominant eigenvalue presence
        - symmetric so all eigenvectors are orthogonal
    - add the ranksource vector in between → interesting
- Dangling link
    - Page with no outgoing link
        - Where there weight should be distributed (as forward link missing)
    - They don't impact others ranking
        - Can be removed to calculate ranking of mainstream pages
        - Can be added later on.
- Implementation
    - Stanford WebBase
    - Webcrawling and indexing system
    - Pagerank Implementation
        - 3.1
        - Reduced computation than indexing
- Convergence Property
    - Expander graph
        - Graph random walk
        - Expander factor connected with eigen values
            - 1st eigenvalue lambda1>>lambda2 (2nd) → matrix have good expansion factor
    - Contain dominant eigenvalues
        - if lambda1>>lambda2, faster converge
        - Power iteration method

- Searching with pagerank
    - Google
    - Title Searching
    - Rank Merging → Difficult task
    - Some sample results
    - Importance on the precision
    - Common case
        - Does well in common searching
        - Subcomponent of common case
    - Personlized pagerank
        - E vector modification
        - problem: related links receive high ranking!!
- Applications
    - webtraffic, navigation,
    - backlink predictor → solves the earlier backlink issues
- Conclusion
    - Everypage → a single number (pagerank)
    - web information retrieval → personal web extraction

10 Wang, F-Y., Zeng, D., Carley, K. M., & Mao, W. (2007). Social Computing: From Social Informatics to Social intelligence. IEEE Intelligent Systems, 22(2), 79-83.

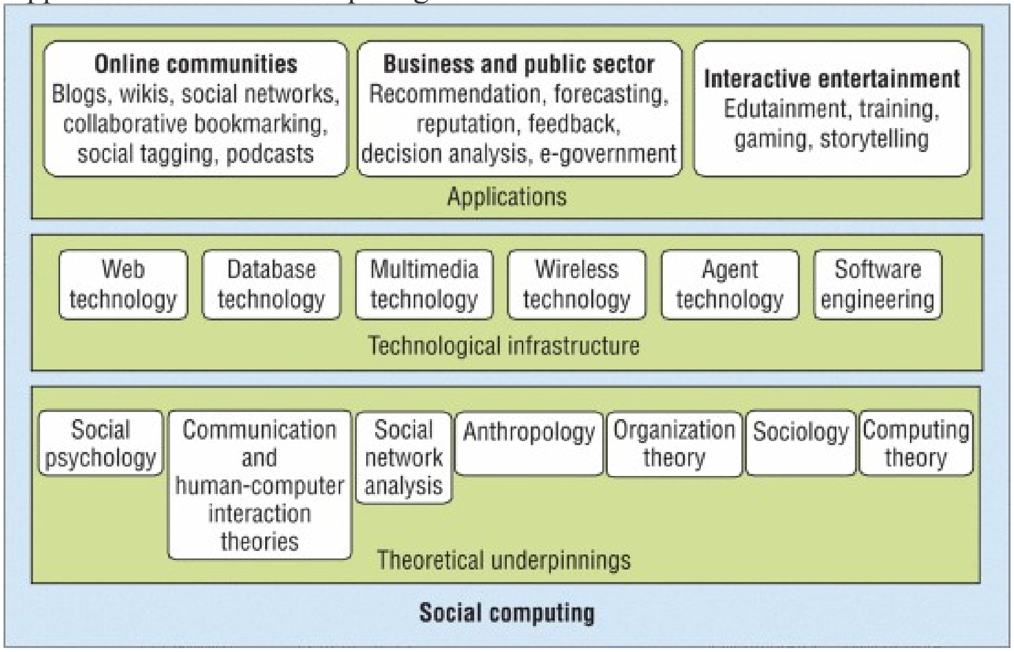
- Computing paradigm and an interdisciplinary research
    - influences system and software development
    - beyond social information processing towards emphasizing social intelligence
        - by capturing human social dynamics,
        - by creating artificial social agents
        - generating and managing actionable social knowledge
- Definition of Social Computing



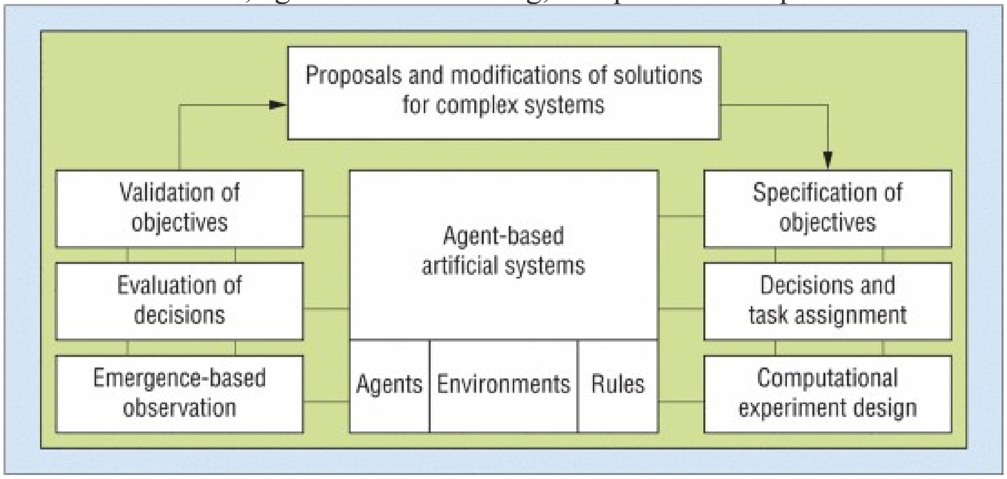
Source	Definition
Communications of the ACM[4]	Describing any type of computing application in which software serves as an intermediary or a focus for a social relation
Wikipedia (http://en.wikipedia.org/wiki/Social_computing, as of December 2006)	Referring to the use of social software, a growing trend in ICT usage of tools that support social interaction and communication
Forrester Research[5]	A social structure in which technology puts power in individuals and communities, not institutions
Our definition[6]	Computational facilitation of social studies and human social dynamics as well as the design and use of ICT technologies that consider social context

-

- Application of Social Computing


Online communities
Blogs, wikis, social networks, collaborative bookmarking, social tagging, podcasts
Business and public sector
Recommendation, forecasting, reputation, feedback, decision analysis, e-government
Interactive entertainment
Edutainment, training, gaming, storytelling
Applications
Web technology
Database technology
Multimedia technology
Wireless technology
Agent technology
Software engineering
Technological infrastructure
Social psychology
Communication and human-computer interaction theories
Social network analysis
Anthropology
Organization theory
Sociology
Computing theory
Theoretical underpinnings
Social computing

- Artificial Societies, agent-based modeling, computational experiments


Proposals and modifications of solutions for complex systems
Validation of objectives
Evaluation of decisions
Emergence-based observation
Agent-based artificial systems
Agents
Environments
Rules
Specification of objectives
Decisions and task assignment
Computational experiment design

- application drives
  - develop better social software to facilitate interaction and communication among groups of people (or between people and computing devices)
  - computerize aspects of human society,
  - Forecast the effects of changing technologies and policies on social and cultural behavior.
- Application area (4)
  - Computer supported online communities
  - Intelligent entities in interactive environments
  - Business & public center forecast
- Research issue
  - Representing social information and knowledge
  - Agent-based social modeling
  - Analysis and prediction

11 Gueorgi Kossinets and Duncan J. Watts (2006), Empirical Analysis of an Evolving Social Network, Science, 311 (5757), 88-90.

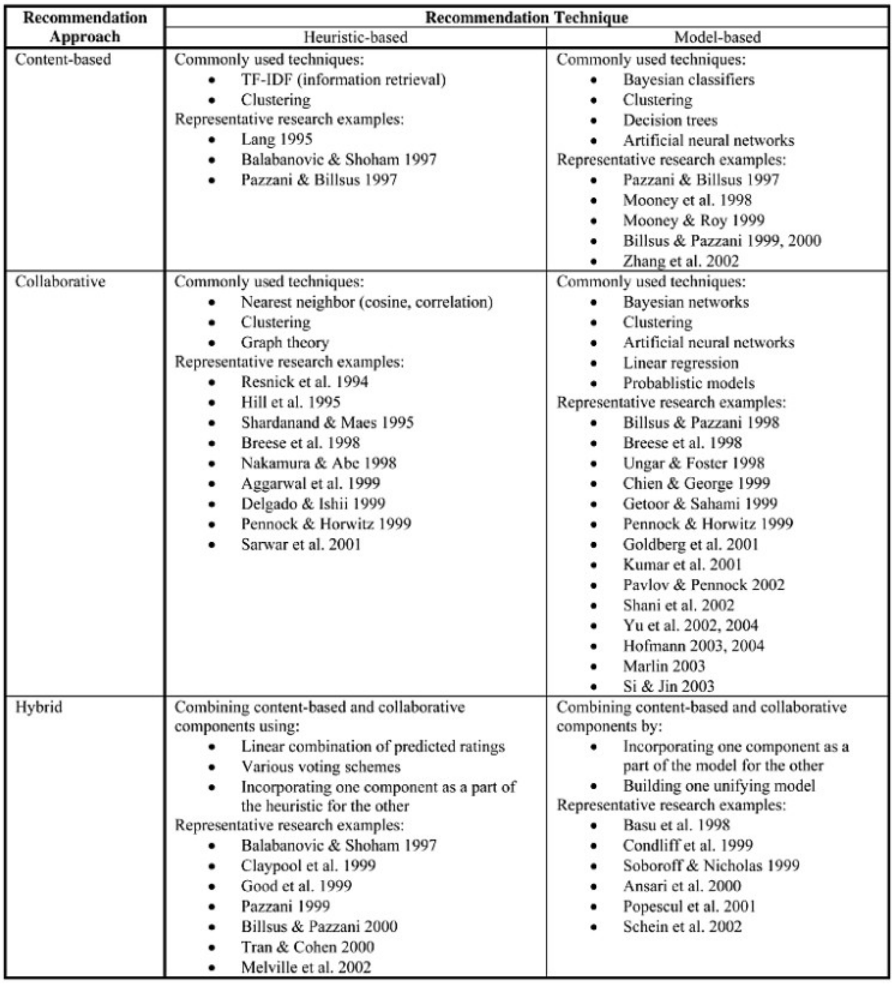
- Empirical analysis with different parameters
- Social network evolve (?? *what does it means*!!)
  - driven by shared activities and affiliations
  - Experiment on: 43,553 students
  - dominated by a combination of effects arising from network topology itself and the organizational structure in which the network is embedded.
  - Average network properties appear an equilibrium state
  - Individual properties are unstable
- Social networks
  - information processing
  - diffusion of social influences
  - distributed search
  - dynamic process
  - individual create-deactivate ties: altering the social structure they participate
  - Complex
  - homophily: interact with similar persons
  - avoiding conflicting relationship
  - exploring acquaintance : Cross cutting
  - Locally dense cluster: mutual benefits : tradic closure: Mutual friends
- Empirical analysis: since complexity
  - longitudinal data: collection over time (rare: in social media)
    - This paper analyzed this longitudinal data
    - created by merging three distinct data structures
      - registry of email (faculty, student, stuffs) – 1year (everything but content)
      - some population – specifying the personal attributes
      - class attended (teacher – student lists)
        - Privacy preserved
    - Email communication – underlying network
      - regularities in expectation
      - Interpersonal communication
      - 14M messages – after preprocessing- removal
    - ongoing relationship: spike of emails
      - Node weight: rate of email transaction w_{ij}
      - strongest path length d_{ij} and the shared affiliation s_{ij}
      - New connections
        - cyclic closure biases (just 3 nodes are connected to each other)
          - Generalize the notion of tradic closure (cycle of three)
        - focal closure biases (one focus)
          - Probability of new ties
        - another one is membership closure
    - d_{ij} increases new connection probability decreases – trivial
    - mutual acquaintance increases new connection probability increase – trivial
    - shared class increase p of new connection increases – trivial
    - Class (faculty, student, stuff, etc) has less impact on connection

- recently cross-sectional data analysis
  - this paper also do some analysis
  - Steady state analysis
- metrics
  - All get seasonal peaks w.r.t TAO (day)
    - Average vertex degree <k>
    - fractional size of the largest component S
    - mean shortest path L
- Explanation of the empirical findings
- Conclusion
  - understanding tie formation processes in social networks requires longitudinal data on both social interactions and shared affiliations
  - With the appropriate data sets, theoretical conjectures can be tested and conclusions previously based on cross-sectional data can be validated.

12 Adomavicius, G. and Tuzhillin, A. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Transactions on Knowledge and Data Engineering. 17(6), June 2005, 734-749.

- Three Recommendation System survey
  - Content based (recommend to similar items liked in the past)
  - Collaborative filtering (recommend items that people with similar taste like)
  - Hybrid filtering (Hybrid of collaborative and content filters)
- This paper offers - Limitation and future works (Extensions)
  - Extensions: Aims to better understand user and Item, support multicriteria rating and more flexible and more intrusive type of recommendation.
- Others
  - Preference based filtering – Predicting relative preference of the users (Alternate of the rating based filter) – not focus of this paper
  - Rating based filter – Focus of this paper
    - approximate rating for new items for a user and recommend based on that.
    - utility function → utility function
    - Defined for limited CxS spaces !!
      - Need extrapolation for the rest
- Content Based Methods
  - Information retrieval/cognitive science, approximation theory root
  - More on item profiling - (TF-IDF)
  - content based user profile, content profile
  - Limitation
    - Limited Content analysis
    - Overspecialization – not recommend new tastes!!
    - New user Problem
- Collaborative methods
  - How to find the C (similar groups) Who liked similar items
    - Two types:
      - Memory based

        - Neighborhood & Graph
      - Model based
        - Clustering
        - Matrix factorization (hidden factor model)
        - Probabilistic methods
    - cluster based approach: Like minded people are clustered together
    - Bayesian Network: bayesian node. structure of the network and the conditional probabilities are learned from the data
  - Find users Similar to a user and recommend.
  - Similarity between vectors of the actual user-specified ratings.
  - Similarity metric modification
    - *default voting*
    - i*nverse user frequency*
    - *case amplification*
    - *weighted-majority prediction*
  - Limitation
    - what if One user single cluster!!
    - New user problem
    - new item problem
    - Sparsity
  - Extension → Model-memory based model combination → performs better than both
  - model based
    - k-means cluster and gibbs sampling
    - probabilistic relational model
    - Maximum entropy model, linear regression
    - Markov Decision process
    - Mixture models
- Hybrid Model
  - Implement collaborative and content-based separately and combine their prediction
  - Incorporate collaborative characteristic into content-based
    - Dimensionality reduction
  - Incorporate content-based character into collaborative
    - collaboration via content
  - General unifying model incorporating both content and collaborative characteristic
    - Single rule-based classifier
- Extensions
  - Comprehensive understanding of users and items
  - Model-based recommendation extensions – Radial basis function
  - Multidimensional instead of 2D (user, items)
    - Why only CxS dimension! → Can extend it
  - Multi-criteria rating.
  - Nonintrusiveness
  - Flexibility
  - Effectiveness of recommendation

- 

Recommendation Approach	Recommendation Technique	
	Heuristic-based	Model-based
Content-based	Commonly used techniques: • TF-IDF (information retrieval) • Clustering Representative research examples: • Lang 1995 • Balabanovic & Shoham 1997 • Pazzani & Billsus 1997	Commonly used techniques: • Bayesian classifiers • Clustering • Decision trees • Artificial neural networks Representative research examples: • Pazzani & Billsus 1997 • Mooney et al. 1998 • Mooney & Roy 1999 • Billsus & Pazzani 1999, 2000 • Zhang et al. 2002
Collaborative	Commonly used techniques: • Nearest neighbor (cosine, correlation) • Clustering • Graph theory Representative research examples: • Resnick et al. 1994 • Hill et al. 1995 • Shardanand & Maes 1995 • Breese et al. 1998 • Nakamura & Abe 1998 • Aggarwal et al. 1999 • Delgado & Ishii 1999 • Pennock & Horwitz 1999 • Sarwar et al. 2001	Commonly used techniques: • Bayesian networks • Clustering • Artificial neural networks • Linear regression • Probablistic models Representative research examples: • Billsus & Pazzani 1998 • Breese et al. 1998 • Ungar & Foster 1998 • Chien & George 1999 • Getoor & Sahami 1999 • Pennock & Horwitz 1999 • Goldberg et al. 2001 • Kumar et al. 2001 • Pavlov & Pennock 2002 • Shani et al. 2002 • Yu et al. 2002, 2004 • Hofmann 2003, 2004 • Marlin 2003 • Si & Jin 2003
Hybrid	Combining content-based and collaborative components using: • Linear combination of predicted ratings • Various voting schemes • Incorporating one component as a part of the heuristic for the other Representative research examples: • Balabanovic & Shoham 1997 • Claypool et al. 1999 • Good et al. 1999 • Pazzani 1999 • Billsus & Pazzani 2000 • Tran & Cohen 2000 • Melville et al. 2002	Combining content-based and collaborative components by: • Incorporating one component as a part of the model for the other • Building one unifying model Representative research examples: • Basu et al. 1998 • Condliff et al. 1999 • Soboroff & Nicholas 1999 • Ansari et al. 2000 • Popescul et al. 2001 • Schein et al. 2002

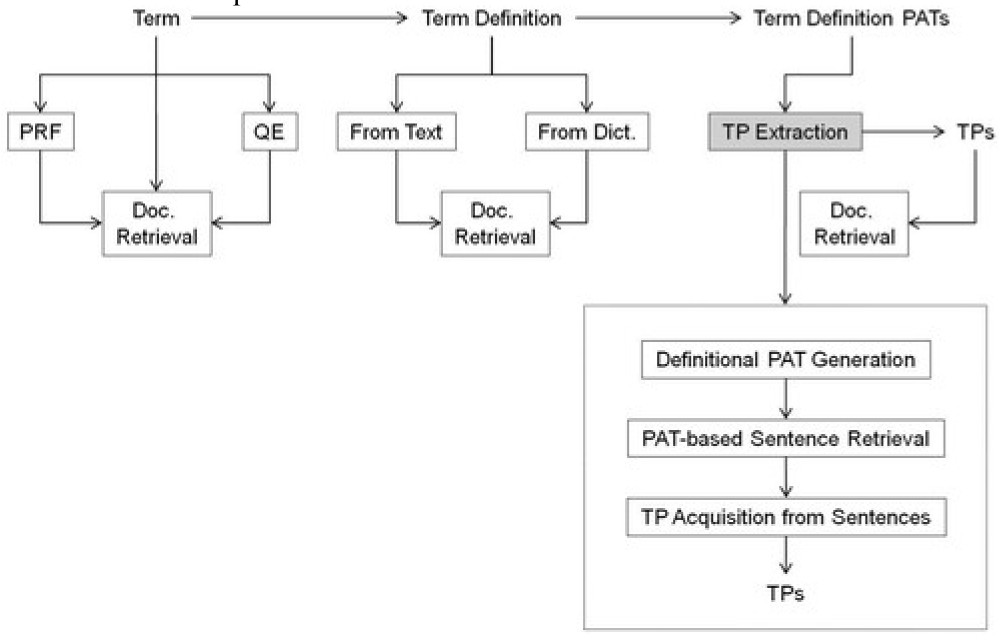
13 Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. Computer, 42(8).

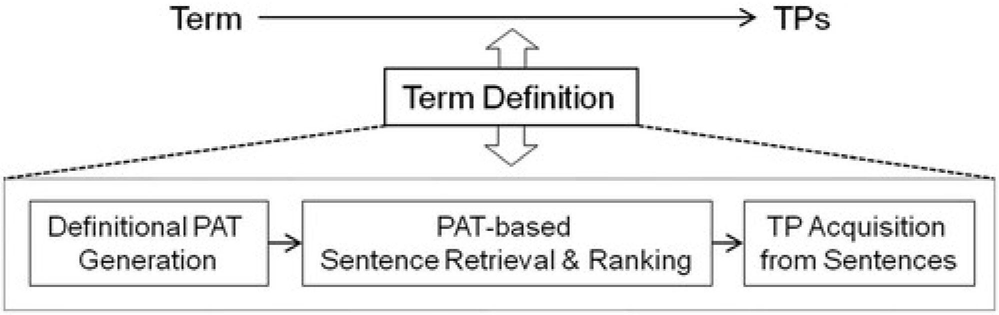
- Netflix prize competition
- Matrix factorization allows
  - Implicit feedback, temporal effects, confidence level
  - Memory efficient, data aspect integration
- Recommended system
  - Content filtering : profile for product / user → products' profile
    - Matching profiles
    - Music Genome project
  - Collaborative filtering: Customer Relation with the products (2 types)
      - Find user-item association (domain free)
      - Better than content filter (but not for new items – cold start problem)
      - better than content-based in general
    - *Neighborhood* methods (relationship between users – item oriented)
      - Figure 1: user-oriented neighborhood method

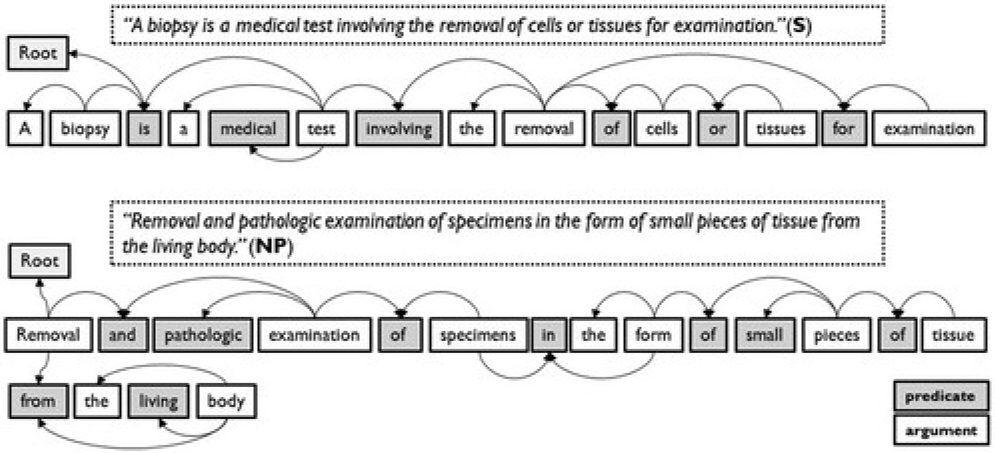
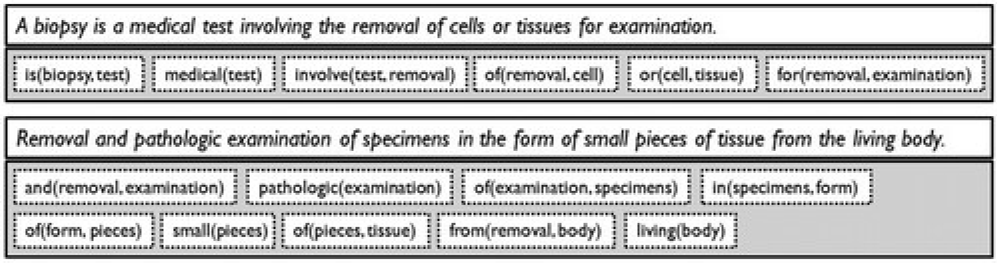
        - ***Latent* factor models** (characterize both users and items) – latent factors discovery → may result uninterpretable dimensions
            - Figure 2: simplified illustration of dimensions and movie projections
            - **Matrix Factorization** (most dominant method)
- Matrix factorization: Scalablility, flexibility, implicit feedback (absence of explicit feedback (the best): providing ratings): densely filled matrix
    - overcome the SVD and missing value issues
    - Represented as latent factor
    - Optimization problem: 2 training algorithms
        - SGD
            - Mostly used
            - Fast
        - alternating least squares
            - Fix one and solve for others → iterative computation
            - Applicable in parallel computation → One independent of others
            - non-sparse training set (centered on implicit data)
    - Flexible to data aspects integration
        - Bias – add users, item and general bias
            - increases complexity by adding new variables to estimate
        - Additional Input Sources : overcome cold start problem
            - item preference
            - attribute information
        - Including temporal dynamics
            - item popularity changes
            - users bias towards item changes in overall
            - users behavior dynamics to a item changes
        - inputs with varying confidence
            - Changes in optimization stage
            - Error multiplied by confidence
            - weights of observed rating → What if adversarial system??
            - probabilistic approaches!
            - So confidence weights on rating estimation errors

14 Sung-Pil Choi and Sung-HyonMyaeng (2012) Terminological Paraphrase Extraction from Scientific Literature Based on Predicate Argument Tuples. Journal of Information Science, 38(6) 593–611.

- TP: sentence/phrases express concept in different form
    - Heuristics/rule based approach
- Predicate-argument tuples: (PATs) Def: A semantic unit, retrieve sentences that contain a terminological concept : Not a rule/heuristic based methods
    - Effective textual similarity computation
    - Based on 6 TP ranking model (frequency, ID, term defin, text, matched, and PMR)

- Two experiments:
    - Newly discovered TPs → Potential of extensible application
- Technical Terms - Key concept
    - Identification, analysis and application of technical terms
    - queries, keyword matching, information retrieval
    - extract semantic equivalent terms
    - 

    - Pseudo-relevant feedback (PRF) and Query Extraction (QE)
- Related works
    - Textual semantic relatedness model
        - Logical relation between texts
    - Paraphrase recognition
        - Equivalence of two texts
    - Term definition (description) extraction
        - Definitional phrase/sentences from texts
    - Query term expansion with controlled vocabularies
        - term mapping methods
- This paper: Terminological paraphrase extraction method
    - PAT based sentence retrieval
    - 
Term
TPs
Term Definition
Definitional PAT Generation
PAT-based Sentence Retrieval & Ranking
TP Acquisition from Sentences
    - Predicted augmented tuples
        - Graph structure: syntactic and semantic relations between words in a sentence

- 
"A biopsy is a medical test involving the removal of cells or tissues for examination."(S)
Root
A
biopsy
is
a
medical
test
involving
the
removal
of
cells
or
tissues
for
examination
"Removal and pathologic examination of specimens in the form of small pieces of tissue from the living body."(NP)
Root
Removal
and
pathologic
examination
of
specimens
in
the
form
of
small
pieces
of
tissue
from
the
living
body
predicate
argument
- predicate-argument structure (PAS) → to PAT
    - PAT : element of PAS and extracted from PAS
    - Syntactic ways to get PAT from PAS
- argument-predicate syntactic relations
    - PAT ratio match (PMR)
- 

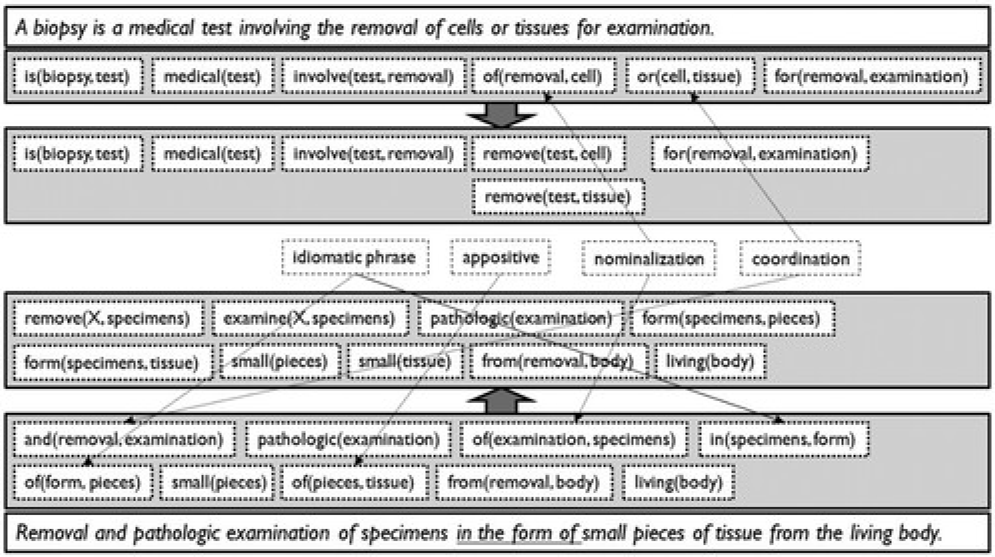
  A biopsy is a medical test involving the removal of cells or tissues for examination.

  is(biopsy, test) medical(test) involve(test, removal) of(removal, cell) or(cell, tissue) for(removal, examination)

  Removal and pathologic examination of specimens in the form of small pieces of tissue from the living body.

  and(removal, examination) pathologic(examination) of(examination, specimens) in(specimens, form) of(form, pieces) small(pieces) of(pieces, tissue) from(removal, body) living(body)

○ Preprocessing of the PATs

- Extraction of PAT and their normalization (used NOMLEX)
- 

  A biopsy is a medical test involving the removal of cells or tissues for examination.

  is(biopsy, test) medical(test) involve(test, removal) of(removal, cell) or(cell, tissue) for(removal, examination)

  is(biopsy, test) medical(test) involve(test, removal) remove(test, cell) for(removal, examination) remove(test, tissue)

  idiomatic phrase appositive nominalization coordination

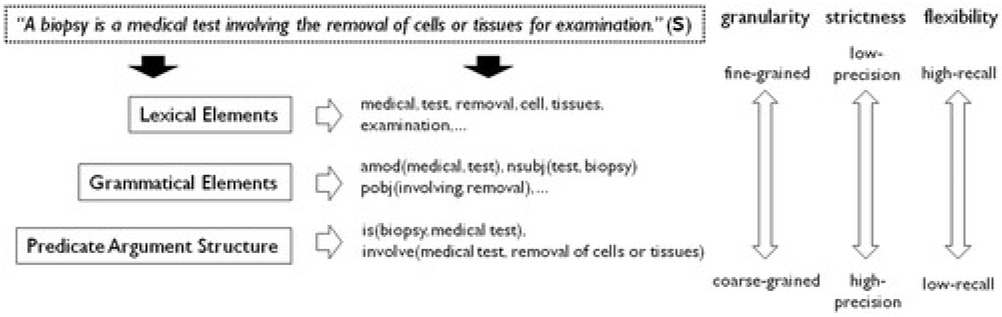
  remove(X, specimens) examine(X, specimens) pathologic(examination) form(specimens, pieces) form(specimens, tissue) small(pieces) small(tissue) from(removal, body) living(body)

  and(removal, examination) pathologic(examination) of(examination, specimens) in(specimens, form) of(form, pieces) small(pieces) of(pieces, tissue) from(removal, body) living(body)

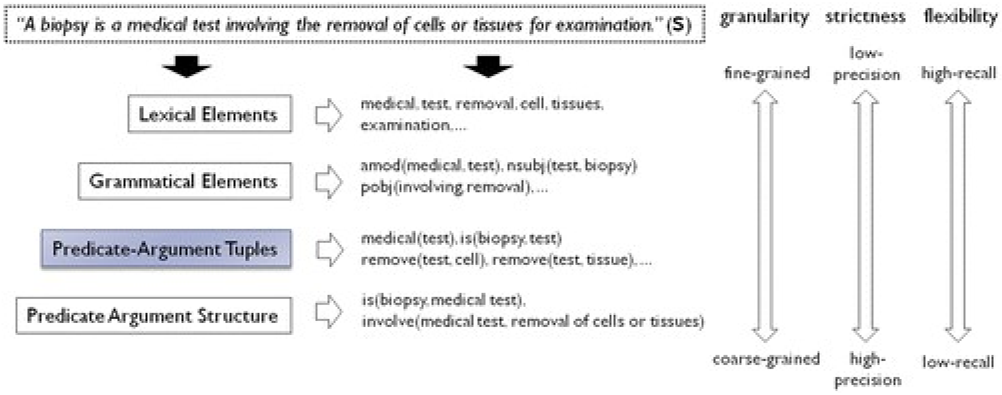
  Removal and pathologic examination of specimens in the form of small pieces of tissue from the living body.

○ PAT-based sentence retrieval

- PAT-based ranking in perspective

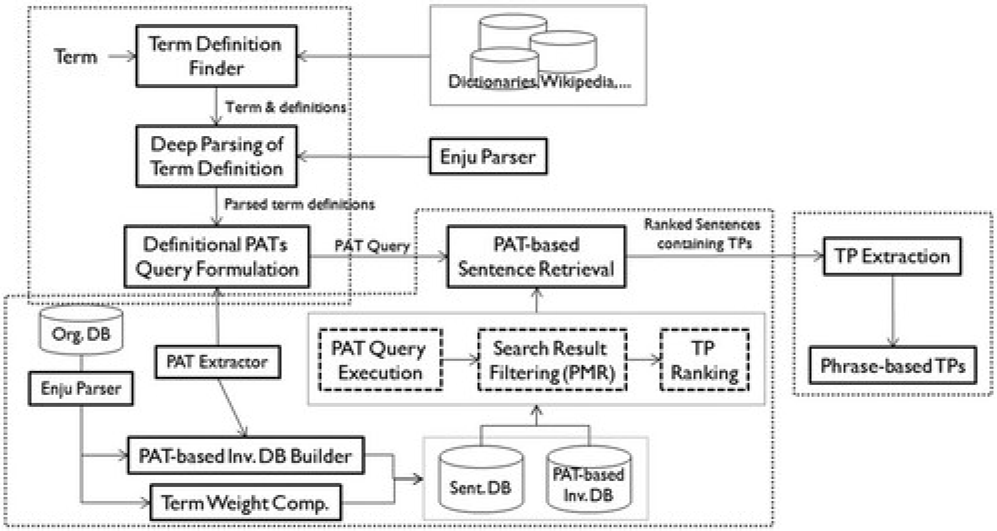

"A biopsy is a medical test involving the removal of cells or tissues for examination." (S)
granularity
strictness
flexibility
fine-grained
low-precision
high-recall
Lexical Elements
medical, test, removal, cell, tissues, examination, ...
Grammatical Elements
amod(medical, test), nsubj(test, biopsy) pobj(involving, removal), ...
Predicate Argument Structure
is(biopsy, medical test), involve(medical test, removal of cells or tissues)
coarse-grained
high-precision
low-recall

- ■
  - Three semantic elements


"A biopsy is a medical test involving the removal of cells or tissues for examination." (S)
granularity
strictness
flexibility
fine-grained
low-precision
high-recall
Lexical Elements
medical, test, removal, cell, tissues, examination, ...
Grammatical Elements
amod(medical, test), nsubj(test, biopsy) pobj(involving, removal), ...
Predicate-Argument Tuples
medical(test), is(biopsy, test) remove(test, cell), remove(test, tissue), ...
Predicate Argument Structure
is(biopsy, medical test), involve(medical test, removal of cells or tissues)
coarse-grained
high-precision
low-recall

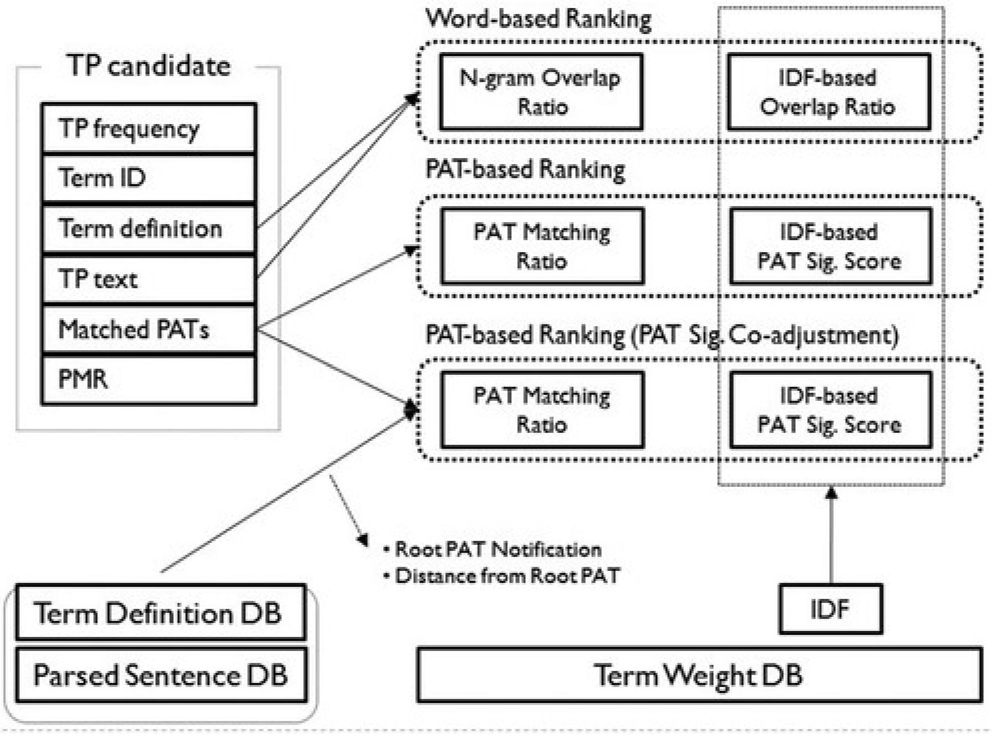
- ■
- ■ Four Semantic Elements

○ Overall system architecture



- ■

○ TP based ranking strategies (six)



        -
        - PAT significance
- Experiments
    - Data: Biomedical domain – PubMed
    - Evaluation of TP ranking methods
        - Term based marco-average
        - Micro-average precision
- Future works and conclusion
    - Extract TP
        - Sentence retrieval system
        - six TP ranking models
    - Requires enhancing performance

15 Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. Proceedings of the National academy of Sciences, 101(suppl 1), 5228-5235.

- Content of documents: Which topics the document addresses
    - Generative Model for document – inspired by Blei, A. Ng et al.
        - Unsupervised approach
    - Formation of Topics: Chose a topic → select/generate words based on it
        - each document is produced by choosing a distribution over topics, and then generating each word at random from a topic chosen by using this distribution.
    - This paper present MCMC algorithm to solve LDA
- Why generative
    - reduces the complex process of producing a scientific paper to → a small number of simple probabilistic steps

  - specifies a probability distribution over all possible documents
  - observed data (the words) are explicitly intended to communicate a latent structure (their meaning)
    - Motivated from *latent dirichlet allocation*
- Application of algorithm
    - Extract topic after analyzing abstract from PNAS using bayesian model selection.
    - Capture meaningful data structured→ Consistent with the authors provided labels
    - Finding hot topics: by
      - Examining temporal dynamics
      - Tagging abstract to illustrate semantic content.
- extract representation – by statistical methods – first order (!) approximation of knowledge in the documents. (unsupervised methods!)
  - Statistical Generative Models → automatic topic findings
- MCMC (this paper)
  - Markov chain to converge a target distribution
    - sample taken from markov chain
    - chain is assigned based on the sampled value
  - used *gibbs sampling* algorithms – heat bath! (A MCMC algorithm)
    - requires full conditional distribution
- Latent variable modeling
    - why not EM
      - Slow convergence and local maxima!!
    - Alternatively, LDA (Key motivation)
      - Variational Bayes (Benchmark 1)
        - Mean field, independence assumptions
        - goes for reverse/backward KL divergence minimization KL(q||p) → mode
      - Expectation propagation (Benchmark 2)
        - leverage the factorization structure of target distribution
        - Goes for forward KL divergence minimization KL(p||q) → mean seeking
      - Consider prior for Theta!! - A dirichlet distribution
        - Conjugate prior for multinomial distribution
    - This work (Consider Posterior of topics for given words!!) then parameters
      - Prior belief changes based on data (change in posterior)
    - Solve by Markov Chain Monte Carlo (this papers contribution)
      - Sample from Constructed Markov Chain
      - Gibbs Sampling (proposed new alg)
- Nice Graphical exampled→ Compared with two (VB, EP) methods.
- Model Selection
  - Depends on three hyperparamers
    - Dirichlet parameters α, β (kept fixed)
    - Model topic number T (Varied) → how many topics

  - T selection → Model section
    - Topic number vs P(w|topic number) plot helps to find the optimal topics (latent variables, may be uninterpretable)
  - Scientific topics and classes
    - Authors chose 3 major categories and 33 minor categories
    - the topics recovered by our algorithm are purely a consequence of the statistical structure of the data – latent structure
  - Hot and Cold Topics : recognize the topics' rise & fall in amount of scientific interest
  - Tagging abstract : assignments of words to topics.
- Conclusion
  - Presented a statistical inference algorithm for LDA
  - Generative approaches
  - application on insight into contents of scientific documents
    - explore topic dynamics
    - WWW
  - Future research
    - Explore complex dynamics / sophisticated algorithm
    - Discover meaningful trends

16 A L Berger, S A Della Pietra, V J Della Pietra (1996). A Maximum Entropy Approach to Natural Language Processing. Computational Linguistics, 1996.

- *Essential parts: understand the difference between the model and the training data!!*
- Statistical model based on maximum entropy (via maximum likelihood)
  - Maximum likelihood approach for constructing maximum entropy models
    - How to implement such approach? - answer this paper
  - Determine the statistics that capture the behavior of random process (feature selection)
  - Given the stats predict the output / future (model selection)
- Statistical modeling
  - constructing model to predict behavior of random process
  - samples, incomplete state of knowledge about the process
    - incomplete knowledge to representation of the process
      - Predict the future based on the representation
- Two essential tasks
  - determine statistical sets → capture the behavior of random process
    - *Feature* Selection (delta function with the features)
  - combine these sets into a accurate model → inference

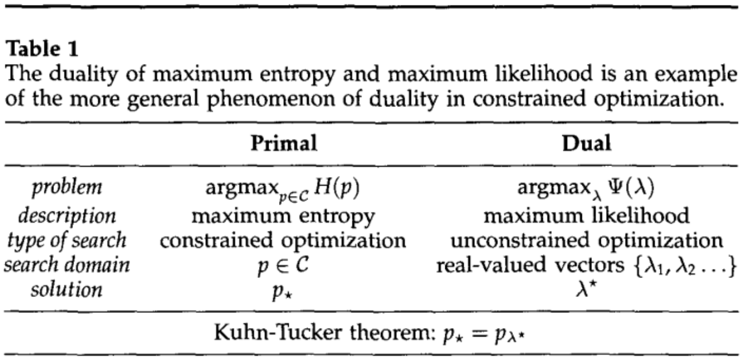
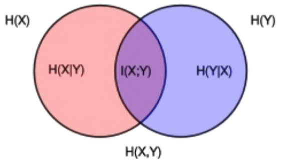
        - *Model* Selection
- maximum entropy and uniformity
    - Proved by Jensen inequality (very easy)
- Uniform behavior for both the train & test data!!
    - Covers maximum information
    - Under different equal option select the uniform one!
    - Agrees with everything known and avoids assuming what is not known
- Maximum entropy modeling
    - *Training data should match the model estimation*
    - Statistics, features, and Constraints
        - Goal: construct model that generated p(x,y) – empirical observation
        - feature: binary valued function
        - *Constraint*: A equation between expected value of feature function in training data and model [eq 3] – key goal
            - Model should correctly occupy the training data
    - Maximum entropy principle
        - To select a model from a set of allowed probability distributions, choose the model with maximum entropy
    - Parametric form → Primal to dual problem
    - Relation to maximum likelihood
        - The model with *maximum entropy* is the model in the parametric family that *maximizes the likelihood* of the training sample.
            - Maximize entropy for over the dataset (known and unknown)
            - Maximize the likelihood over the known observations
        - 

Table 1
The duality of maximum entropy and maximum likelihood is an example of the more general phenomenon of duality in constrained optimization.

	Primal	Dual
problem	$\text{argmax}_{p \in \mathcal{C}} H(p)$	$\text{argmax}_{\lambda} \Psi(\lambda)$
description	maximum entropy	maximum likelihood
type of search	constrained optimization	unconstrained optimization
search domain	$p \in \mathcal{C}$	real-valued vectors $\{\lambda_1, \lambda_2 \ldots\}$
solution	$p_\star$	$\lambda^\star$

Kuhn-Tucker theorem: $p_\star = p_{\lambda^\star}$

    - Computing the parameters → Algorithm 1
- Feature Selection (2 steps)
        - finding appropriate facts
        - Combining the facts into the model
    - Basic feature selection → Algorithm 2
    - Approximate gain
        - Feature ranking
        - models that maximizes the approximate gain.

        - 

- Case studies
    - review of statistical translation
    - context dependent word models
    - segmentation & word reordering
- Conclusion
    - building block for maximum entropy modeling
    - Chose the model that has maximum entropy
    - optimal values of these parameters are obtained by maximizing the likelihood of the training data
    - the model with the greatest entropy consistent with the constraints is the same as the exponential model which best predicts the sample of data.
    - Propose algo for constructing maximum entropy model
        - selecting features → usually very slow
        - computing parameters of the model containing the features
    - Applications → context-sensitive modeling

17 M. Simpson, and D. Demner-Fushman (2012) Biomedical Text Mining: A Survey of Recent Progress. C.C. Aggarwal and C.X. Zhai (eds.). Mining Text Data, Springer.

- Text mining technologies
    - Overview of current technologies
        - Emphasis on resource and tools
        - relation events
        - Major task (with basic challenges and influential works)
            - implicit fact discovery
            - Summarization
            - question answering
- intro
    - regularly reviewed due to fields growth
    - application
        - acceleration in discovery
        - timely access to materials
        - Relation among facts
    - requires: Biologists and clinicians
    - bibliome (the entirety of the texts relevant to biology and medicine)
        - Step wise approaches
    - Information retrieval
    - Database: MIMIC II, ORBIT

  - Discussed:
    - tokenization
    - Parts of Speech tagging
    - parsing
- Resource of biomedical text mining
  - corpora
    - Medicine, Medical subject heading (MeSH)
  - Annotation
    - guidelines by other literature
    - three approaches
      - human annotators knowledge
      - assisted annotation (output of annotation tool is corrected)
      - ontology based
  - Knowledge sources
    - rich set of supports NCBO
  - Supporting tools
    - based upon the UMLS is MetaMap
- Information extraction
    - process by which structured facts are automatically derived from unstructured or semi-structured text.
    - BioNLP
    - BioCreAtivE
  - Named entity recognition (NER)
    - same names around
  - Relation extraction
    - Relation between entities
  - Event Extraction
    - nested event
- Summarization
  - process by which the salient aspects of one or more documents is identified and presented succinctly and coherently
  - ROUGE, PERSIVAL
- Question Answering
    - process of providing direct and precise answers to natural language questions.
  - Medical Question answering
  - Biological Question answering
- Literature based discovery
  - Definition: the task of utilizing scientific literature to uncover “hidden,” previously unknown or neglected relationships between existing knowledge.
  - Goal: identify relations worthy of further scientific investigation or to find evidence supporting suspected relations
- conclusions
  - event extraction and clinical text mining
  - increasing the public availability of and community investment
  - development and use of common frameworks, such as UIMA

18 LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton (2015). Deep Learning, Nature 521(7553), 436-444.

- Focus on CNN
- Basics and backpropagation
- Future works
  - Representation learning with complex reasoning

19 C J C Burges (1998). A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery. 2, p. 121-167. Available at http://research.microsoft.com/pubs/67119/svmtutorial.pdf

- Contents
  - Concept of VC dimension
  - Structural Risk minimization (general) → we want lower VC confidence
    - Empirical risk minimization (on train data only)
  - SVM for separable and non-separable data
    - when solution is unique or global?
      - Unique solution: linearly separable for fixed number of support vectors
      - Quadratic optimization: Convex problem → Local solution is the global
    - Kernel mapping?
    - Connection to VC dimension?
      - When infinite VC dimension?
      - VC dimension of polynomial and the radial-basis function
    - Generalization performance → Higher VC dimension is bad
        - can be shown in Structural risk minimization bound
- Drive behind SVM development
  - Bias variance tradeoff: Right balance, may
    - (high bias – underfit), low variance
      - too stiff model, high bias about certain model properties
    - Low bias – overfit, high variance
      - model change by changing training dataset
  - Capacity control: learn from data without error : overfit-underfit
  - Overfitting:
- A bound for generalization
  - Vapnik-1995 generalization bound
  - VC dimension
    - Concept of shattering – $\{f(\alpha)\}$ member can correctly assign label
      - any combination of label is possible in shattering cases
    - VC dimension h means for at least one set of h points the function family can shatter → assign any labeling combination
    - (h) – is maximum (no $h+1$ can be shattered by VC dimension of h)
    - VC dimension of R^n hyperplane is (n+1) - Maximum points
      - choosing one as origin (1) and rest as linearly independent (n) → n+1

  - Capacity of function
  - Shattering depends on points choice
  - VC dimension → Chose points that can be shattered
  - VC dim +1 → can't be shattered anyways!
- Structural risk minimization:
  - Chose function with lowest VC dimension among equal good options

- Linear SVM
  - Separable case – SVM straightforward → Unique solution
  - KKT condition
  - Test phase → find w and project it on it and compare with the bias
  - Non-separable form – Interesting
- Nonlinear SVM
  - kernel
  - Test phase case → Equation 61
  - Mercer's Condition → Existence of kernel (gram matrix)
  - Global Solution and Uniqueness
    - Every local solution is global
      - Convex programming problem property
      - in not strictly convex then intermediate solutions are also global solution
    - Unique – if the objective function is *strictly* Convex→ hessian matrix PD: pos def
  - Unique (if Dual Function is convex function)
  - SVM training always finds a global solution is in contrast to the case of neural networks, where many local minima usually exist
- VC dimension of SVM
  - if map to H space then VC dimension is |H|+1.
    - |H| is the dimension of space H
  - Polynomial Kernel p on data (data dimension of d_L)
    - VC dimension is [{d_L + p -1} Chose {p}] +1
  - Radial basis function the VC dimension is INFINITE
    - Can be generalized for any number of points!!
    - if the kernel matrix is full rank then these classifiers has infinite VC dimension
    - Theorem 5: when distance between different points are large if the K(xi, xj) → 0
      - These kernel has infinite VC dimension
      - radial basis function
- Limitation
  - Choice of kernel → once fixed, live with it
  - Speed and size in both test and training
  - Discrete data !!
  - Multiclass SVM in single step is still not feasible !!
- Conclusion
  - SVM always finds the global minima

20 Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).

- word2vec extensions
    - Non-contextual embedding
        - One word one representation
        - Doesn’t depend on the context/sentence
        - language models embedding depends on the sentence/context
    - CBOW (cont. bag of words)
    - Continuous Skip Gram model
- Skip gram model
    - learn good representation → Capture syntactic and semantic relationship
- This paper: Extension of Continuous skip gram model (neg and subsampling)
    - Improve quality of representation → more regular representation
    - Faster (Subsampling of frequent words)
        - Alternative of hierarchical softmax → negative sampling: NCE variant
    - Good vector representation of words or **Phrases** → includes phrase
    - Vector-like calculation over the representations
- distributed word representation → Similar words together
- The skip-gram model
    - one word to multiple words
    - impractical to compute for all the words
        - Alternative: Hierarchical softmax
            - How to form the tree structure
            - Binary Huffman tree
        - This Paper: Negative Sampling
            - NCE: Ranking above to differentiate data from noise
            - investigate different choice for negative distributions
        - This paper: Subsampling frequent words
            - Capture co-occurrence of words [France with Paris]
                - Not the common co-occurrence [the with every words]
            - counter the imbalance between rare and frequent words
            - High frequent words are discarded more
- Empirical results
    - Tasks 2
        - Syntactic analogies (quick:quickly :: slow:slowly)
        - Semantic analogies (country to capital)
    - NEG performs better than NCE
    - subsampling improves the training speed and makes more accurate
    - Non-linearity in network improves the overall results
- Learning Phase
    - Trained on google dataset (1 billion words! ~682K vocab size)
    - Phrase considered by data driven approach – Bigram model
        - Find score (> th then phase!)
    - Phrase skip gram model results
        - showed in paper

- Additive Compositionality → volga river ~ russian river
- Comparison → Faster and accurate representation
- conclusion
  - how to train distributed representations of words and phrases with the Skip-gram model
  - representations exhibit linear structure, makes precise analogical reasoning possible.
  - Possibly extension to CBOW
  - Better representation for uncommon words & faster
  - Hyper-parameters selection is task-specific
  - crucial decisions that affect the performance
    - model architecture,
    - the size of the vectors, t
    - he subsampling rate, and
    - the size of the training window. (skip gram words)
- Input output word embedding [answer 2] (https://stats.stackexchange.com/questions/263284/what-exactly-are-input-and-output-word-representations)
- More about representation (https://towardsdatascience.com/word-embeddings-exploration-explanation-and-exploitation-with-code-in-python-5dac99d5d795)
- More about hsoft and neg samp (https://towardsdatascience.com/hierarchical-softmax-and-negative-sampling-short-notes-worth-telling-2672010dbe08)

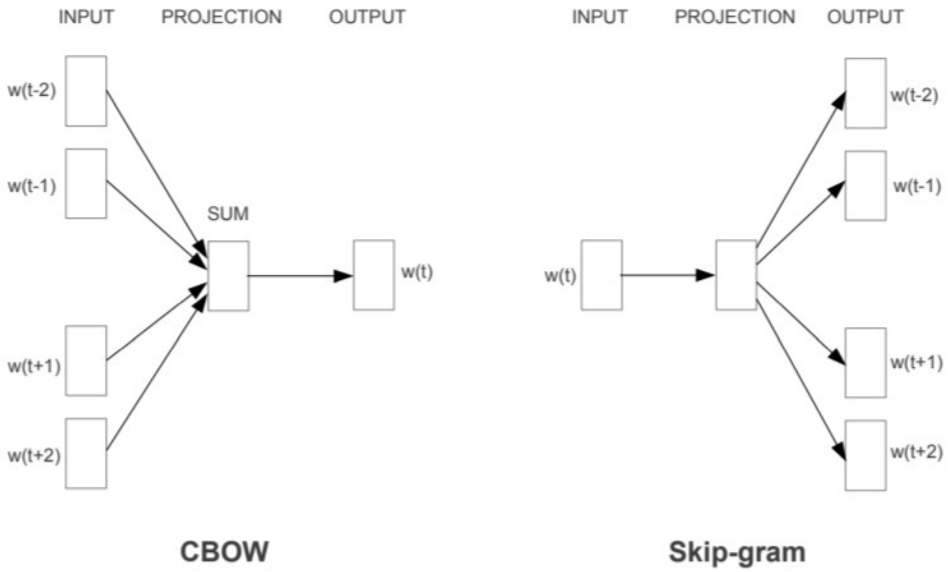


Figure 1: Source: https://arxiv.org/pdf/1309.4168v1.pdf

21 Ghahramani, Z., & Jordan, M. I. (1997). Factorial Hidden Markov models, Machine Learning, 29, pp. 245–273

- HMM – time series probabilistic model learning
- Generalization of HMM
  - Factored into multiple states
  - More distributed way

- Exact Algorithm – to infer the posterior prob of hidden variables
  - Connection to forward-backward algorithm
  - connection to graphical model
- Approximate inference
  - Gibbs Sampling/ Variational methods
    - Variational model- Decouple the variables
    - Structural approximation for FHMM
  - FHMM capture statistical structures (HMM failed those)
- HMM
  - Representation of past state!
  - Distributed State representation
- *This paper: Efficient learning algorithm for HMM with distributed state representation.*
- Motivation
  - state space to feature decomposition, decouple the dynamic process
  - Distributed representation simplifies the problem
- Motivated from work of Hinton et al
- HMM with distributed state – Graphical model
  - Node – Hidden state
  - Connection – Conditional distribution
-

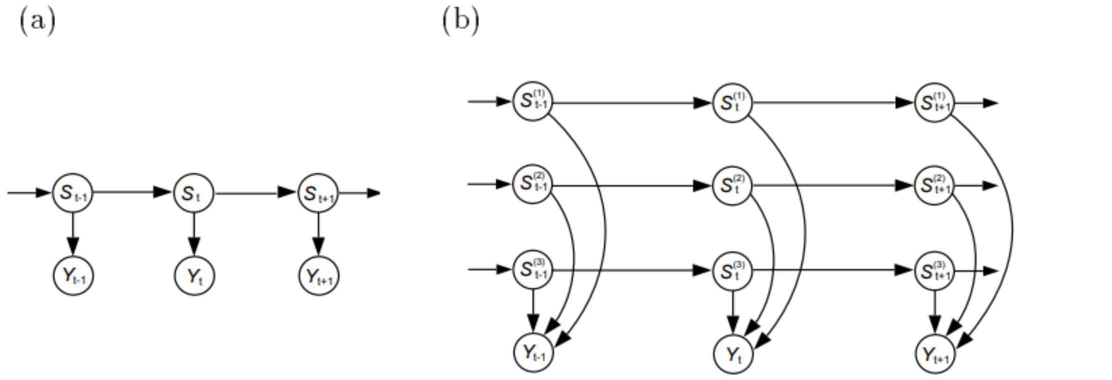


Figure 1. (a) A directed acyclic graph (DAG) specifying conditional independence relations for a hidden Markov model. Each node is conditionally independent from its non-descendants given its parents. (b) A DAG representing the conditional independence relations in a factorial HMM with $M = 3$ underlying Markov chains.

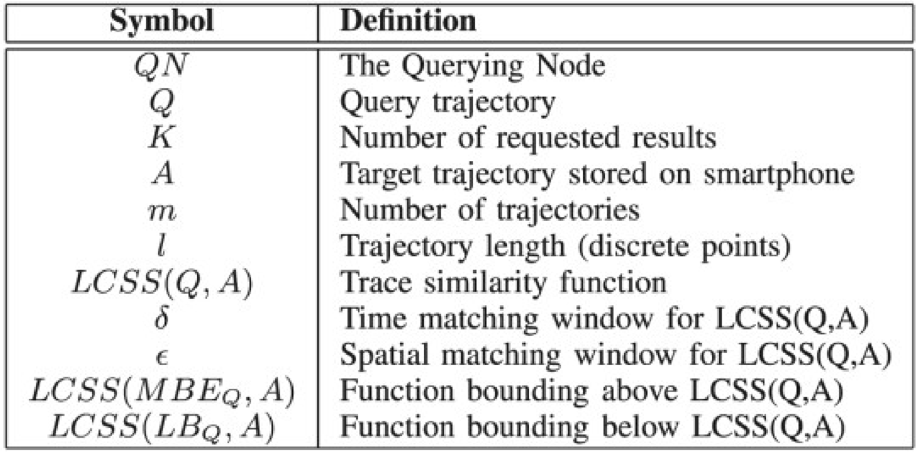
- Probabilistic Models
- This paper: state be a collection of state variables - FHMM
  - Constrained on the state transition matrix - figure (b)
  - State variable has its own dynamics
  - proposed new model
- Inference and Learning
  - EM algorithm
    - E-M steps by breaking the log likelihood
  - Exact inference
    - Intractable

- Inference: Gibbs Sampling
    - initialization
    - Given the observation
    - Neighbor state examination
    - Requires 1st and 2nd order statistics
- Completely factorized variational inference
    - factorized the parameters
- Structured variational inference
    - tractable
    - connected to ELBO

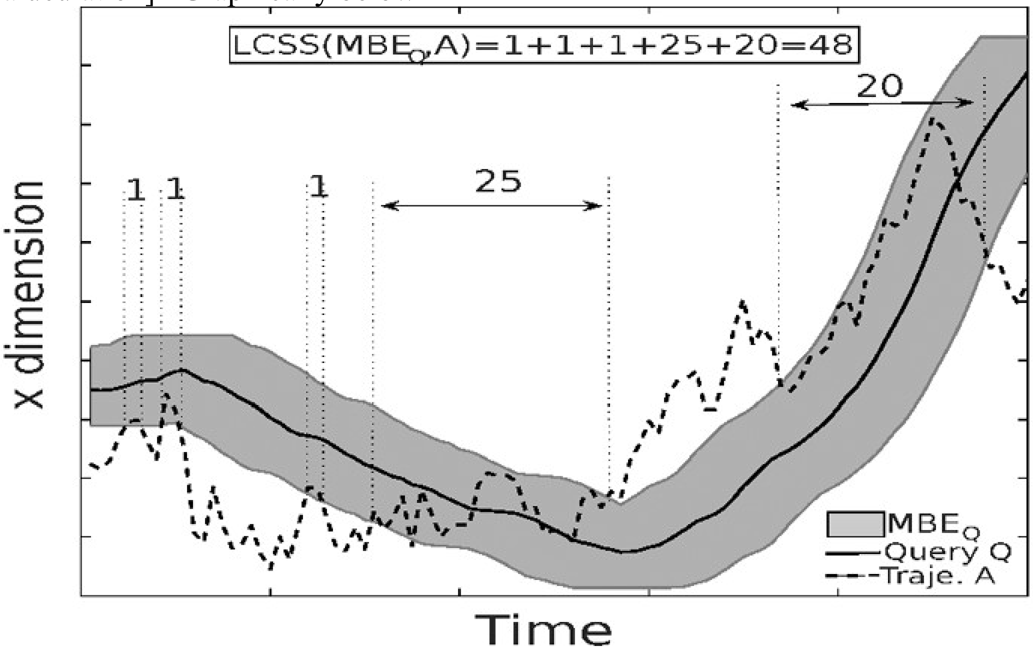
- Total *m chain* all together: HMM has single chain
- Very nice idea, multiple chain instead of single chain (just that!)
- Inference
    - Computing prob of hidden variables | observation
        - forward-backward algorithms
    - Finding Most probable output
        - Viterbi Algorithm
- Learning problem
    - Learning structures of the model
        - A problem in ML and graphical modeling
    - Learning parameters
        - This paper solves: given the structure for FHMM
- Experiments (5 Coomparisons)
    - HMM by Baum-welch algorithm
    - Factorial for E step and forward-backward
    - fHMM using gibbs
    - fHMM using complete factorization
    - fHMM using structured Variational approximation
- Generalization of fHMM
    - Introduce coupling
        - interconnection between the chain states → 2nd order statistics
    - Conditioning on the input
    - Hidden markov decision tree

22 Demetrios Zeinalipour-Yazti, Christos Laoudias, Constandinos Costa, Michail Vlachos, Maria I. Andreou, Dimitrios Gunopulos (2013). Crowd sourced Trace Similarity with Smartphones, IEEE Transactions on Knowledge & Data Engineering, vol.25, no. 6, pp. 1240-1253, June 2013

- Problem target:
    - compare query trace Q against a lot of traces stored and generated on distributed smartphones
- Technologies used: Mobile computing service in distributed system
- Propose: smartTrace+
    - *crowdsourced trace similarity* search framework

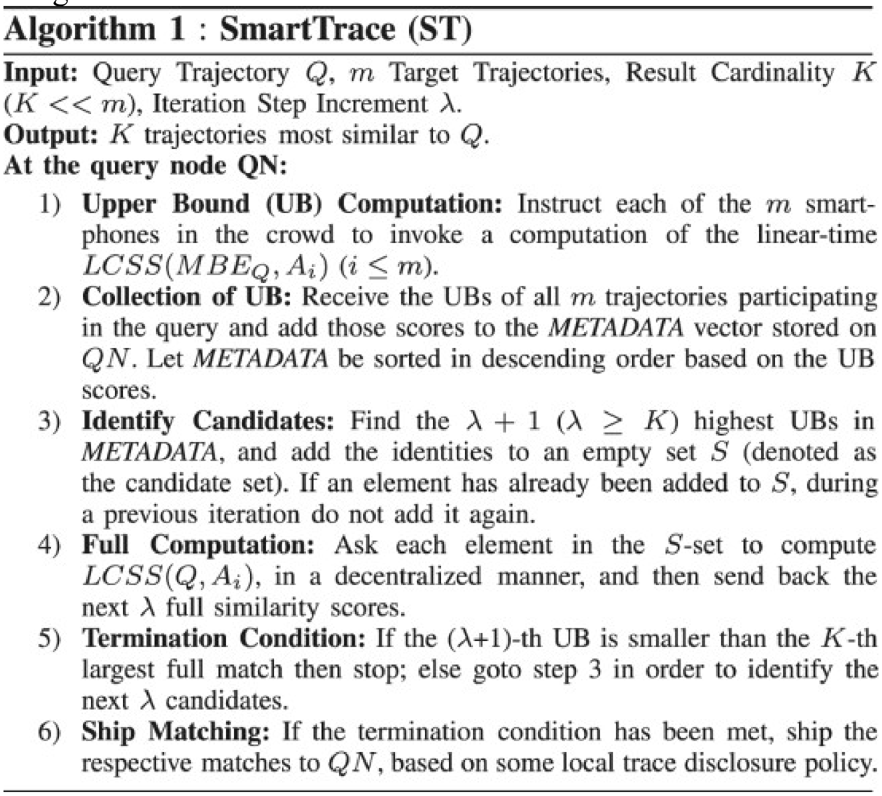
  - Considered privacy
  - in-situ data storage
  - top-K query processing exploiting distributed trajectory
  - tested on testbed – 25 phones
    - synthetic and real workloads
  - Benefits
    - Resilient to temporal and spatial noise.
    - Energy conservation.
    - Faster
- Crowd-sourcing → Distributed problem solving method
  - Active participation of users
- Traditional Related works
  - Centralized trajectories search
  - Match score with stored trajectories in cloud
  - take advantages of global knowledge structure
  - privacy issues → participant data is reveal
  - power issues → constantly sending data to server
- Proposed new framework smartTrace+
  - Find users that move more similar to Q, where Q is query trace
  - More on privacy - Asking the users to participate
    - without disclosing the traces of participating users to the querying node
  - Decentralized → Looses the centralized advantage, However, avoids
    - Efficiency of communication links
    - Disclosing personal data to central entity
  - Arrange tuple of trajectories in same time and space frames
  - Apply top-k processing algorithm
  - Proposes two algorithms
    - Offer iterative SmartTrace (ST) algo → iterative method
    - non-iterative StamtTrace (NIST) algorithm
      - works in two phases
      - reduce response time
      - However, increases networking traffic
  - Analytic study of performance and convergence property of algorithm.
  - Prototype development → google OS
  - 

Symbol	Definition
QN	The Querying Node
Q	Query trajectory
K	Number of requested results
A	Target trajectory stored on smartphone
m	Number of trajectories
l	Trajectory length (discrete points)
$LCSS(Q, A)$	Trace similarity function
δ	Time matching window for LCSS(Q,A)
ϵ	Spatial matching window for LCSS(Q,A)
$LCSS(MBE_Q, A)$	Function bounding above LCSS(Q,A)
$LCSS(LB_Q, A)$	Function bounding below LCSS(Q,A)

- Experimental study
- Hypothesis:
    - Processing and networking are equally expensive.
    - Centralized trajectory comparison takes more energy!!!
- Trajectory Similarity Background
        - Point-to-point Matching
            - Norms and distance → Bad choice (because shifting and scaling!!)
        - Time-shifted matching
            - *Longest Common Subsequence (LCSS)*,
            - he *Dynamic Time Warping (DTW)* (Almost comparable with LCSS)
            - the *Edit Distance on Real Sequences (EDR)* (the worse)
            - the *Edit Distance with Real Penalty (ERP)* (the worse)
        - This Paper: LCSS
            - quadratic time
            - instead use bounding above LCSS – linear time.
- Two LCSS properties bound
    - Upper bound, LCSS(MBEQ,Ai)
    - Lower bound LCSS(LBQ,Ai)
- This Work on LCSS, LCSS(LBQ,Ai) – Linearity in time
    - So LCSS (MBE(Q), A) [upper bound of the next] instead of LCSS (Q, A) [expensive caluculation]– Graphically below
    - 

- SmartTrace Framework
    - ST calculate the upper bound of matching between Q and Trajectory
        - Theoretically showed that ST correctness – return most similar
        - showed convergence analysis Big O (m/lambda) in worst case
    - NIST calculates both Upper bound and Lower bound
        - Theoretic proof for correctedness
        - Both upper bound and Lower bound LCSS(LBq, Ai) calculation – simple approximation of the LCSS(Q, A)

    - Again compare the full after sorting top k
- Store UB and LB as metadata! (original path are the Data)
- k highest matches are returned from the metadata!!

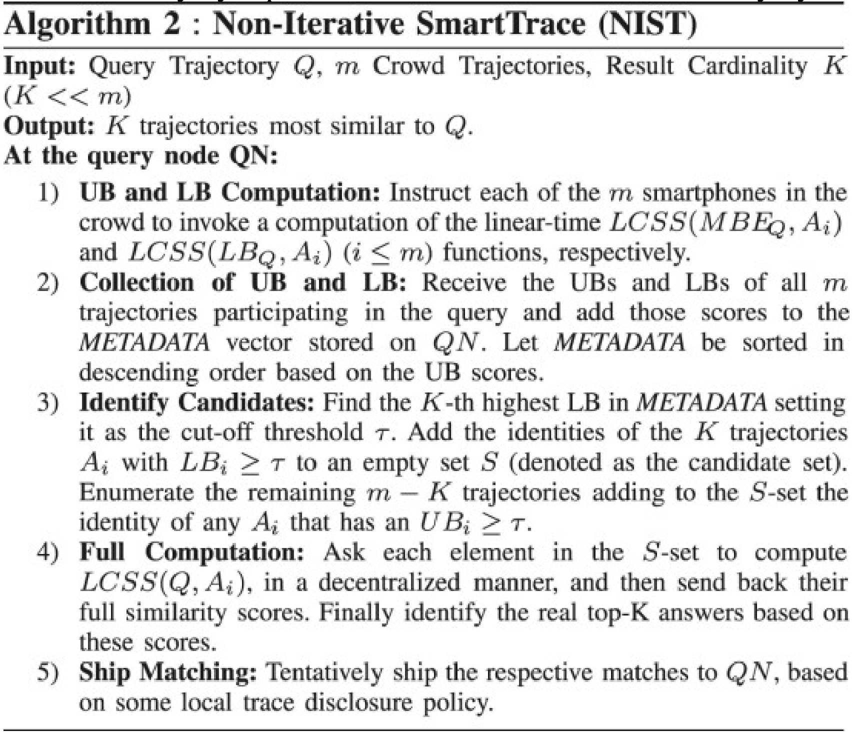


Algorithm 1 : SmartTrace (ST)

Input: Query Trajectory Q, m Target Trajectories, Result Cardinality K ($K << m$), Iteration Step Increment λ.
Output: K trajectories most similar to Q.
At the query node QN:

1) **Upper Bound (UB) Computation:** Instruct each of the m smartphones in the crowd to invoke a computation of the linear-time $LCSS(MBE_Q, A_i)$ $(i \leq m)$.
2) **Collection of UB:** Receive the UBs of all m trajectories participating in the query and add those scores to the *METADATA* vector stored on QN. Let *METADATA* be sorted in descending order based on the UB scores.
3) **Identify Candidates:** Find the $\lambda + 1$ $(\lambda \geq K)$ highest UBs in *METADATA*, and add the identities to an empty set S (denoted as the candidate set). If an element has already been added to S, during a previous iteration do not add it again.
4) **Full Computation:** Ask each element in the S-set to compute $LCSS(Q, A_i)$, in a decentralized manner, and then send back the next λ full similarity scores.
5) **Termination Condition:** If the (λ+1)-th UB is smaller than the K-th largest full match then stop; else goto step 3 in order to identify the next λ candidates.
6) **Ship Matching:** If the termination condition has been met, ship the respective matches to QN, based on some local trace disclosure policy.

- Find similarity by top K UB → then calculate full similarity by LCSS methods



Algorithm 2 : Non-Iterative SmartTrace (NIST)

Input: Query Trajectory Q, m Crowd Trajectories, Result Cardinality K ($K << m$)
Output: K trajectories most similar to Q.
At the query node QN:

1) **UB and LB Computation:** Instruct each of the m smartphones in the crowd to invoke a computation of the linear-time $LCSS(MBE_Q, A_i)$ and $LCSS(LB_Q, A_i)$ $(i \leq m)$ functions, respectively.
2) **Collection of UB and LB:** Receive the UBs and LBs of all m trajectories participating in the query and add those scores to the *METADATA* vector stored on QN. Let *METADATA* be sorted in descending order based on the UB scores.
3) **Identify Candidates:** Find the K-th highest LB in *METADATA* setting it as the cut-off threshold τ. Add the identities of the K trajectories A_i with $LB_i \geq \tau$ to an empty set S (denoted as the candidate set). Enumerate the remaining $m - K$ trajectories adding to the S-set the identity of any A_i that has an $UB_i \geq \tau$.
4) **Full Computation:** Ask each element in the S-set to compute $LCSS(Q, A_i)$, in a decentralized manner, and then send back their full similarity scores. Finally identify the real top-K answers based on these scores.
5) **Ship Matching:** Tentatively ship the respective matches to QN, based on some local trace disclosure policy.

- Discussed the prototype system

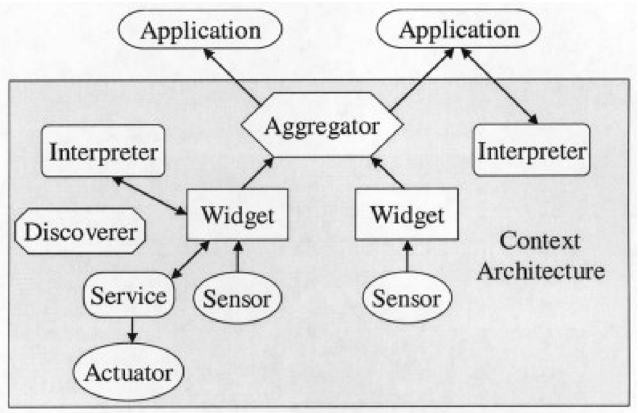
    - GUI
    - Protocol
  - Analytic Evaluation
    - Cost Model
    - Performance analysis
      - Centralized performance (C)
      - Decentralize performance (D)
      - ST performance
      - NIST performance
  - Experiment
    - Methodology
      - Dataset: Oldenburg, GeoLife-A, GeoLife-B
          - both indoor (WiFi Received-Signal- Strength)
          - outdoor (GPS coorinate)
        - Synthetic
        - real-world
      - algorithm and metrics
        - analysis of Energy (T)
        - analysis of time (E)
    - Comparison (C, D, ST, NIST)
    - Ablation study
      - Varying K (size of the answer set)
      - Varying iteration number (lambda)
    - Prototype evaluation
- Conclusion
  - Reach similar result of Centralized and fully decentralized methods
  - Future:
    - Large scale field study of crowd-sourcing services
    - additional metrics!! DTW

23 Anind K. Dey, Daniel Salber and Gregory D. Abowd (2001). A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-aware Applications. Human Computer Interaction. 16 (2-4), pp. 97-166

- ubiquitous computing
- enhance the behavior of any application by informing it of the context of its use
  - Context: related to the interaction between humans, applications, and the surrounding environment.
- 3 main problem
  - notion of context ill defined
  - lack of conceptual models
  - lack of jump-start tools
- This paper: developed context toolkit
  - instantiates this conceptual framework and supports the rapid development of a rich space of context-aware applications
  - demonstrate how such a framework can support the investigation of important research challenges in the area of context-aware computing

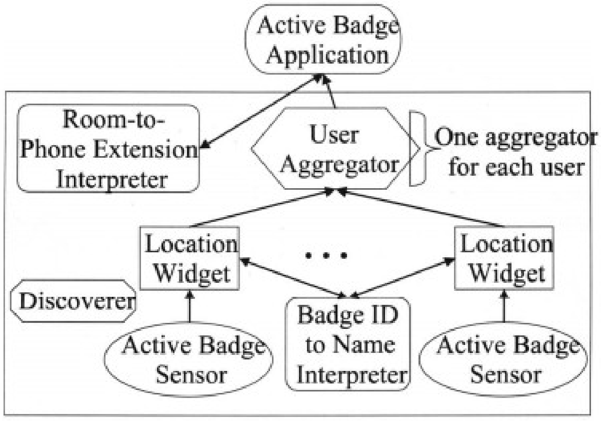
- Introduction
  - context-aware computing
    - Scenario
    - Difficulties in handling context
  - 3 key Goals
    - operational understanding of context
    - conceptual framework to assist in the design of context-aware applications
    - facilitate context-aware computing research by allowing the empirical investigation of the design space and the exploration
- Definition of context
  - : any information that can be used to characterize the situation of entities (i.e., whether a person, place, or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves. Context is typically the location, identity, and state of people, groups, and computational and physical objects.
  - Categories of context (4)
    - identity
    - location
    - status (activity)
    - time
  - Context aware functions (3)
    - Presenting information and services
    - automatically executing services
    - attaching context information for later retrieval
- Requirements for handling context
  - dealing with context
    - • Separation of concerns.
    - • Context interpretation.
    - • Transparent, distributed communications.
    - • Constant availability of context acquisition.
    - • Context storage and history.
    - • Resource discovery.
  - Context abstraction
    - A context widget is a software component that provides applications with access to context information from their operating environment.
    - interpreters,
    - aggregators,
    - services,
    - discoverers
  - using the framework

Figure 1. **Example configuration of Context Toolkit components.**



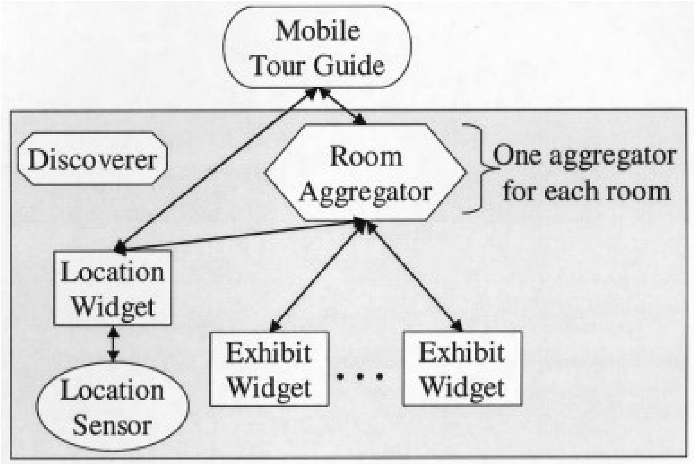
    -
    - Active badge call-forwarding

Figure 2. **Architecture diagram for the Active Badge call-forwarding application.**



    -
    - Mobile tour guide

Figure 3. **Architecture diagram for the mobile tour guide application.**



    -

- Context aware toolkit
    - Implemented for this work

  - Distributed communications

Figure 4. **Object diagram for the Context Toolkit abstractions.**

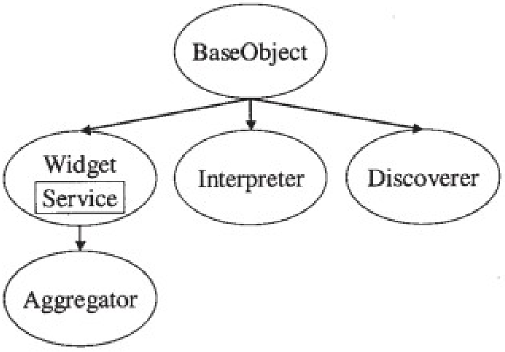
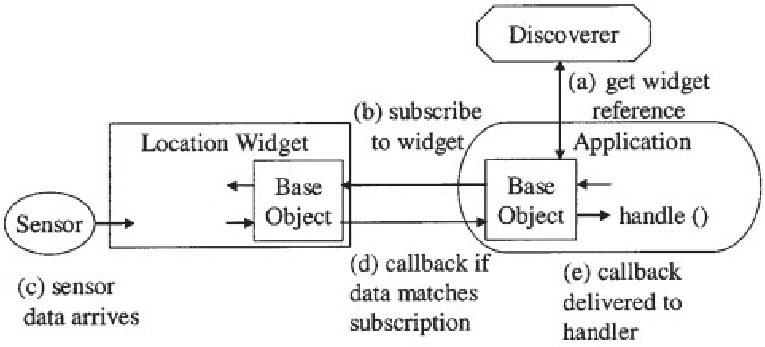


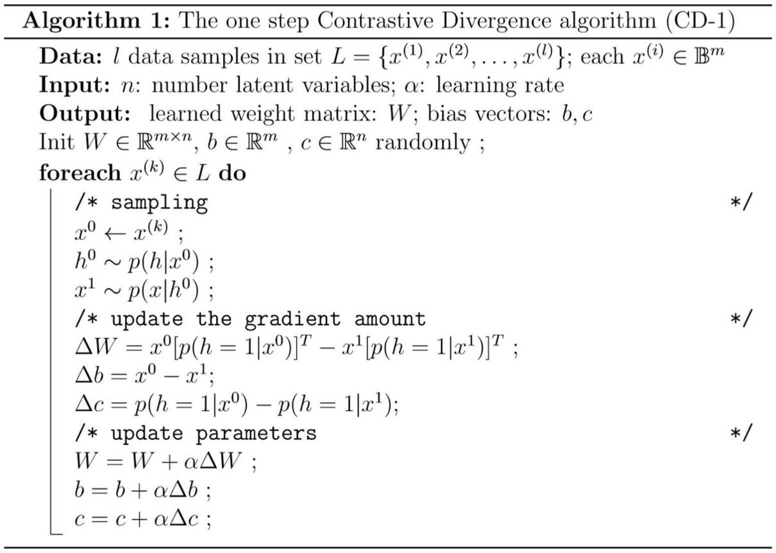
Figure 5. **Application subscribing to a context widget.**



  -
  - Subscription
  - Event handling
- In/out board context aware mailing list
  - Architectures figures
  - intercom applications

24 Philemon Brakel and Dirk Stroobandt and Benjamin Schrauwen, Training Energy-Based Models for Time-Series Imputation, Journal of Machine Learning Research, pp. 2771-2797, 2013

- This paper: training energy-based graphical models for imputation directly
  - Difficult in probabilistic approach
  - inspired from: optimization-based learning, CNN, NN
  - used 3 neural net
    - convolution over data: Described in section 2
    - RNN: Described in section 3
    - Markov random field with hidden state representations: section 4
  - loss gradient
    - Backpropagation through GD
    - Backpropagation through Mean-field
  - truncated GD
    - better than Contrastive divergence algoithm
    - cd algorithms



```
Algorithm 1: The one step Contrastive Divergence algorithm (CD-1)
Data: $l$ data samples in set $L = \{x^{(1)}, x^{(2)}, \ldots, x^{(l)}\}$; each $x^{(i)} \in \mathbb{B}^m$
Input: $n$: number latent variables; $\alpha$: learning rate
Output: learned weight matrix: $W$; bias vectors: $b, c$
Init $W \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ , $c \in \mathbb{R}^n$ randomly ;
foreach $x^{(k)} \in L$ do
    /* sampling                                              */
    $x^0 \leftarrow x^{(k)}$ ;
    $h^0 \sim p(h|x^0)$ ;
    $x^1 \sim p(x|h^0)$ ;
    /* update the gradient amount                            */
    $\Delta W = x^0[p(h=1|x^0)]^T - x^1[p(h=1|x^1)]^T$ ;
    $\Delta b = x^0 - x^1$;
    $\Delta c = p(h=1|x^0) - p(h=1|x^1)$;
    /* update parameters                                     */
    $W = W + \alpha\Delta W$ ;
    $b = b + \alpha\Delta b$ ;
    $c = c + \alpha\Delta c$ ;
```

    - training methods handle the missing values
    - introduces one artificial and two real-world datasets
  - Data
    - meteorology, finance, and physics, are high dimensional time-series,
    - Speech
    - impute missing values
    - noise / malfunction sensors
    - generated by complex nonlinear processes
  - Experimental data
    - USPS handwritten digit data (concentrated hand written)
    - Motion Capture – Conditional RBM
    - missing training data: SCITOS G5 robot navigation room
  - Probabilistic graphical model
    - Generative model- intractable
    - not good, issues with long range dependencies
  - undirected graphs – markov random field
    - intractable, not good
    - EM not good
- needs to capture non-linear dependencies
- Limitation
  - Long training time
  - Hyperparameter tuning
  - difficult model to train
- Benefit of parallel computing

25 Abbasi, Ahmed; Zhang, Zhu; Zimbra, David; Chen, Hsinchun; and Nunamaker, Jay F. Jr.. (2010). "Detecting Fake Websites: The Contribution of Statistical Learning Theory," *MIS Quarterly*, (34: 3) pp.435-461.

- Stat. Learn Theory (SLT) based fake website detection

    - 20% of entire website
    - This paper: discuss challenges, and shortcoming of existing methods
        - Creating new class of fake detection (by design science paradigm)
            - Using STL
    - Develop: AZProtect
    - Empirical demonstration on test bed
- Prototype using design science theory
    - Data: 900 test cites
    - compared with others
- Existing systems
    - look up systems – blacklists of other trading communities
        - May be too late for some early peoples
        - Client-server architecture to maintain blacklist of known fake sites
    - Proactive classification techniques
        - Fraud cues searching
    - Detect fake website - A dynamic process
- Fake Websites
    - Characteristics: Aesthetics, professional looking
    - Categories (2 major)
        - Target search engines (web spam)
        - Target web users (This paper: fraud and monetarily involved)
            - Spoof: copying others like paypal, bofa
            - Concocted: new standalone frauds sites
    - Contain Fraud cues (hypothesis) – 3 majors
            - Website design
            - information navigation
            - visual design
        - Details fraud cues (selected features)
            - Web page text
                - Misspelling, grammatical errors
                - lexical measure
            - Web page source code
                - HTML spoof
            - URLs
                - https, .org, .us, .biz, .info
            - Images
                - prior images
            - Linkages
                - relative links
        - Challenges
            - Inherent cues
- SLT based methods
    - Target: diverse and vast fake site detection, incorporate rich fake cues, contain domain knowledge, sustainability against long-term dynamics adversaries
    - ML approaches
- SLT overview

    - SVM
    - Error bounds
- Application of SLT for the fake website detection
    - generalize
    - rich fraud cues
    - domain knowledge incorporation
    - Dynamic learning
    - AZprotect: rich fraud cue settings
        - 5 features: Discussed in Details fraud cues
    - SVM classifiers (used different kernels)
        - hinge loss
        - linear custom composite performs the best
    - Metrics
        - overall accuracy, classlevel precision, and class-level recall. Additionally, class level f-measure and receiver operating characteristic plots/curves
        - Class-level f-measure is the harmonic mean of precision and recall. ROC plots/curves depict the relationship between true positive and false positive rates
- Hypothesis:
    - should perform better in accuracy, recall, precision – trivial
    - SLT will be the best model
- Evaluation
    - Comparison with lookup and classification methods.
- Future
    - Different hierarchical system for spoof and concocted websites

26 Hevner, A.R., March, S.T., Jinsoo Park, J. and Ram, S. (2004).Design Science in Information Systems Research. MIS Quarterly. 28(1), 75-105.

- IS research:
    - Behavioral ()
        - natural science
        - theories to predict human/organization behavior/interaction \
        - improve efficiency of an organizations
    - **Design science (create new innovative actifacts** to extend human/organization capability). (DS)
        - Engineering and science
        - Problem solve
        - applicability by designing artifacts
            - bounded by natural laws
            - implementable
    - cycle between design and behavioral-science
        - opposite side of same coin
        - kind-a-inseparable
- Objective
    - performance of design-science research
    - Concise framework for understand, executing and evaluating research

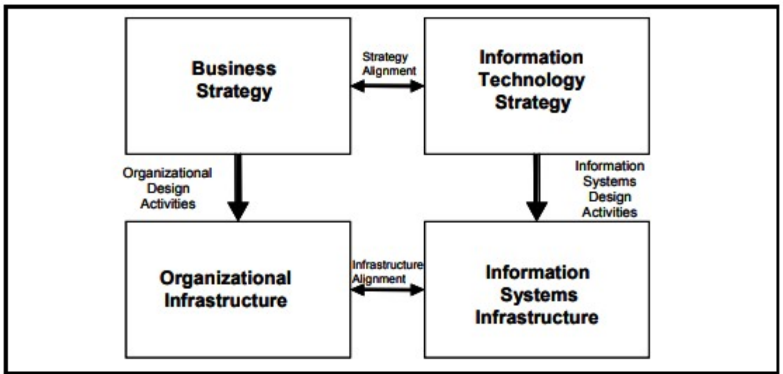
- the primary goal of this paper
  - inform the community of IS researchers and practitioners of how to conduct, evaluate, and present design science research.
  - Accomplished by
    - describing boundaries of design science via conceptual frameworks
    - understand IS research – guideline to conduct and evaluate good DS research
  - Focus on technologies
- Explained with three recent exemplars (application)
  - Group decision support system (GDSS)
  - Exchangable routing language
  - design theory for IS to support emergent knowledge processes
- High quality DS
  - call for synergistic efforts between Behavioral and DS researchers.
- 
Business Strategy
Strategy Alignment
Information Technology Strategy
Organizational Design Activities
Information Systems Design Activities
Organizational Infrastructure
Infrastructure Alignment
Information Systems Infrastructure

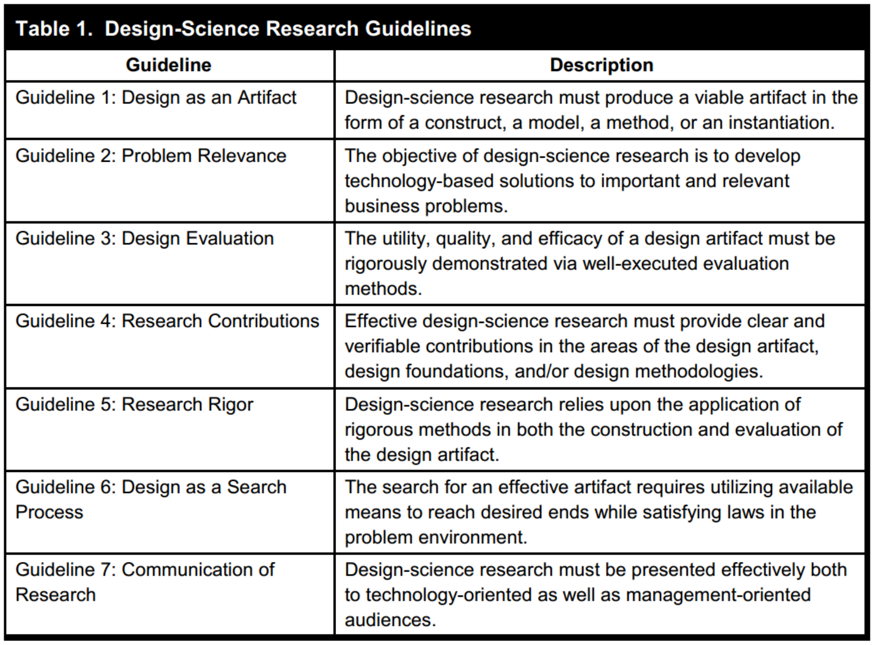
  Figure 1. Organizational Design and Information Systems Design Activities (Adapted from J. Henderson and N. Venkatraman, "Strategic Alignment: Leveraging Information Technology for Transforming Organizations," *IBM Systems Journal* (32:1), 1993.)
- 

Table 1. Design-Science Research Guidelines

Guideline	Description
Guideline 1: Design as an Artifact	Design-science research must produce a viable artifact in the form of a construct, a model, a method, or an instantiation.
Guideline 2: Problem Relevance	The objective of design-science research is to develop technology-based solutions to important and relevant business problems.
Guideline 3: Design Evaluation	The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods.
Guideline 4: Research Contributions	Effective design-science research must provide clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies.
Guideline 5: Research Rigor	Design-science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact.
Guideline 6: Design as a Search Process	The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment.
Guideline 7: Communication of Research	Design-science research must be presented effectively both to technology-oriented as well as management-oriented audiences.

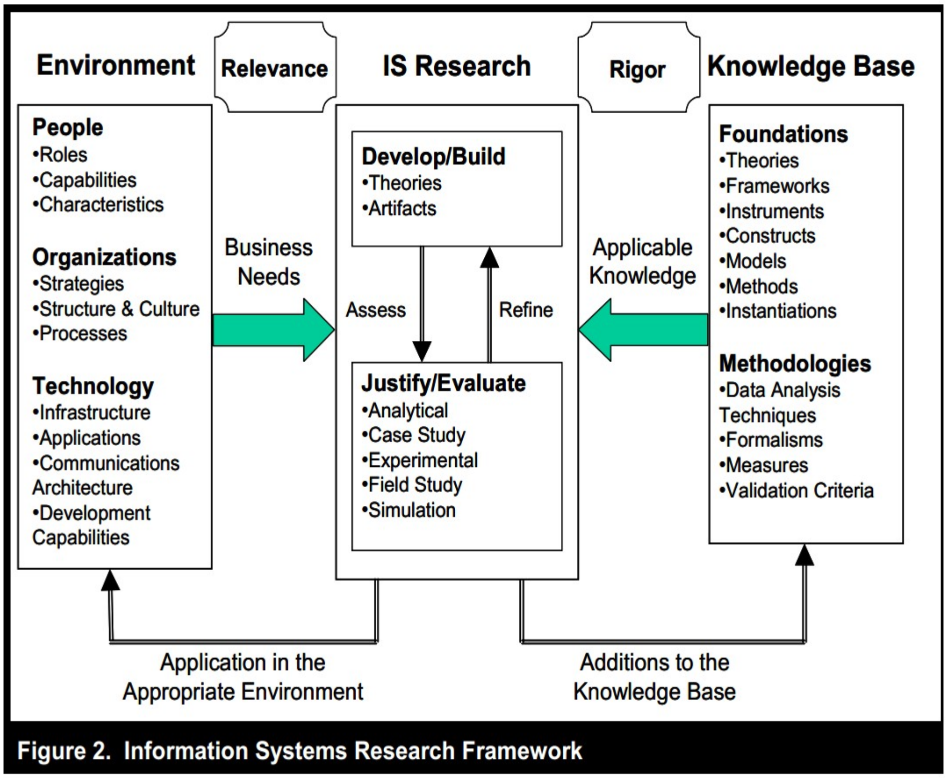


Figure 2. Information Systems Research Framework

Table 2. Design Evaluation Methods

1. Observational	Case Study: Study artifact in depth in business environment
	Field Study: Monitor use of artifact in multiple projects
2. Analytical	Static Analysis: Examine structure of artifact for static qualities (e.g., complexity)
	Architecture Analysis: Study fit of artifact into technical IS architecture
	Optimization: Demonstrate inherent optimal properties of artifact or provide optimality bounds on artifact behavior
	Dynamic Analysis: Study artifact in use for dynamic qualities (e.g., performance)
3. Experimental	Controlled Experiment: Study artifact in controlled environment for qualities (e.g., usability)
	Simulation – Execute artifact with artificial data
4. Testing	Functional (Black Box) Testing: Execute artifact interfaces to discover failures and identify defects
	Structural (White Box) Testing: Perform coverage testing of some metric (e.g., execution paths) in the artifact implementation
5. Descriptive	Informed Argument: Use information from the knowledge base (e.g., relevant research) to build a convincing argument for the artifact's utility
	Scenarios: Construct detailed scenarios around the artifact to demonstrate its utility

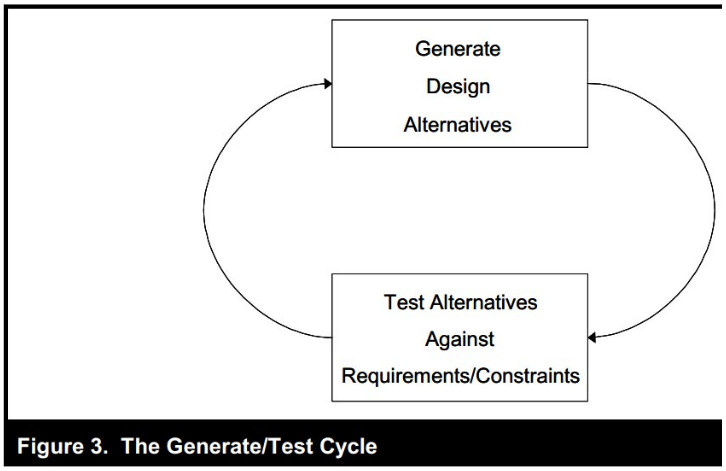
- 

  Figure 3. The Generate/Test Cycle

27 BERNERS-LEE, TIM, JAMES HENDLER, and ORA LASSILA. "THE SEMANTIC WEB." Scientific American 284, no. 5 (2001): 34-43. http://www.jstor.org/stable/26059207.

- Linking between the data itself in web
  - Web 2.0 → The Semantic Web is the web of connections between different forms of data that allow a machine to do something it wasn’t able to do directly
- a new form of web content → Next generation web
- A semantic web
  - Centralized data linking
  - time/location shared across all
  - Personal semantic agents
  - Unified information
- Expressing meaning
  - Semantic web tomorrow
  - currently no semantic → only link sharing
- Semantic web
  - The Semantic Web will enable machines to COMPREHEND semantic documents and data, not human speech and writings
  - Bring order and structure into the web
  - roaming page to page and carry out tasks
    - unified information
    - Semantic meaning of web-pages
  - Extension of current web
  - Computer human better cooperation
  - Machine understand data
  - automatic www processing
  - Decentralized
  - Challenges: (i) provide a language that expresses both data and rules for reasoning (ii) that allows rules from any existing knowledge-representation system to be exported onto the Web
  - Adding logic to web
  - Important technology for developing SW:

        - extensible Markup Language (XML) : Create own tags, hidden labels
            - The basic idea behind the Semantic Web was that everyone would use a new set of standards to annotate their webpages with little bits of XML. These little bits of XML would have no effect on the presentation of the webpage, but they could be read by software programs to divine meaning that otherwise would only be available to humans.
        - resource description framework (RDF): expression of meaning
    - RDF uses :
        - Meaning expressed in set of triplet (sub, verb, obj) → XML tag written
        - sub, obj → a Universal Resource Identifier (URI), just as used in a link on a Web page. (URLs, Uniform Resource Locators,)
    - Ontologies
        - Communicate across database for same information – Combine info.
        - Common meaning across database
        - Third basic components of database: collection of information: Ontology
            - Taxonomy: Object class and their relation
            - Inference rule: supplies further power
        - postal code and address are similar!
        - Enhance Web functions
    - Agent
        - Collect web content from diverse source , process information and exchange result – true semantic web power
        - Agents working together
        - translate internal reasoning of semantic web's unifying language
        - Digital signature: verify from trusted source
        - Check resource
        - Service discovery:
            - Web-based services already exist without semantics, but other programs such as agents have no way to locate one that will perform a specific function
        - Bootstrap new information
        - value chain
- Evolution of knowledge → assist evolution of human knowledge
- Breakout the virtual realm → RDF to service TV, cell phone
- The Semantic Web, in naming every concept simply by a URI,
    - unifying logical language will enable these concepts progressively linked into a universal Web
- WWW → anything can link anything
- knowledge representation
    - traditional knowledge-representation systems generally each had their own narrow and idiosyncratic set of rules for making inferences about their data
    - structure collection of information and conduct automated reasoning
- https://twobithistory.org/2018/05/27/semantic-web.html

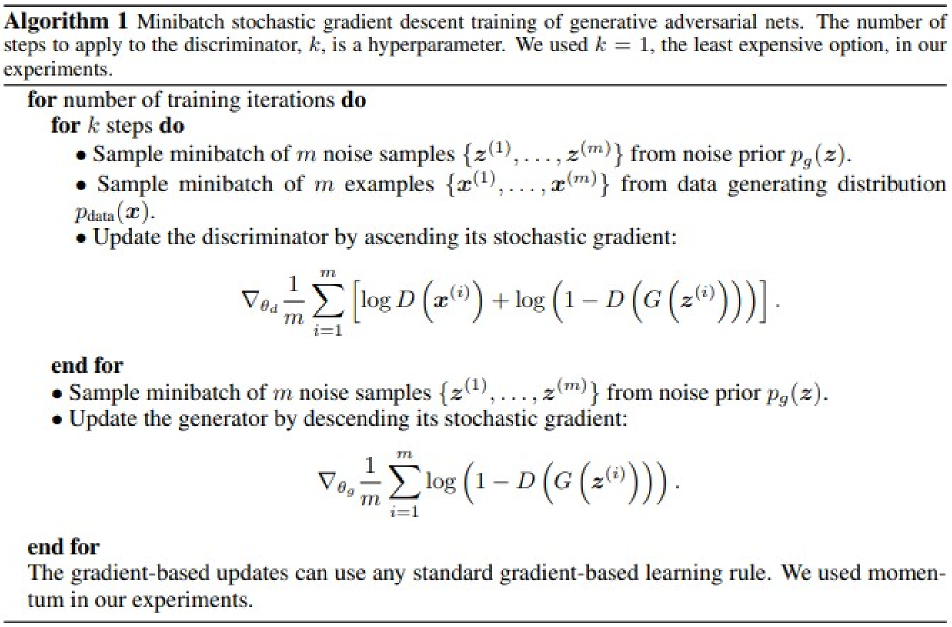
[28] Airoldi, E. M., Blei, D. M., Fienberg, S. E., & Xing, E. P. (2008). Mixed membership stochastic blockmodels. Journal of Machine Learning Research, 9(Sep), 1981-2014.

- pair wise measurement given and application in
    - protein interactions
    - social network
- analysis pairwise connection measurement with probabilistic models
    - independence or exchange ability assumptions no longer hold!!
        - requires special assumptions
- Introduced variance allocation model → mixed membership stochastic block models.
    - combine global parameters (instantiate dense patches of connectivity) (called blockmodel) with local parameters (instantiate node-specific variability in the connections) (called mixed membership)
    - Develop generalized Variation inference algorithm to fast estimate of posterior
    - application: Social media and the protein interaction
- conclusions
    - introduces the Membership stochastic blockmodel
        - novel class of latent variable models for relational data
        - the observations can be represented as a collection of unipartite graphs.
        - variational inference algorithm is parallelizable and allows fast approximate inference on large graphs.

29 Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets." In Advances in neural information processing systems, pp. 2672-2680. 2014.

- Generative model via an adversarial approach
    - Train 2 models (Arbitrary differentiable functions) – Multilayer perceptrons
            - trained by backpropagation
        - Generator G → forces the mistake from D
        - Discriminator D → Tries to separate G outputs
    - 2 player Min-max game
    - Unique solution – G recovers training data → D is 0.5 everywhere
    - No need for Markov Chains or unrolled inference network
    - Both qualitative and quantitative analysis
- DNN success → BP, dropout, piecewise linear unit (nice gradient properties)
- GAN benefits
    - approximate intractable probabilistic computations
    - Leverage piece-wise linear unit in generative context
- Adversarial network → G-D min-max objective
- Related works
    - Directed graphical models with latent variable
    - Undirected graphical models with latent variable
            - Deep Boltzmann Machines & variants
            - Restricted Boltzmann Machines & variants
        - Estimated by MCMC methods like gibbs sampling
            - mixing causes more problems
            - solution requires integration over all the hidden variables

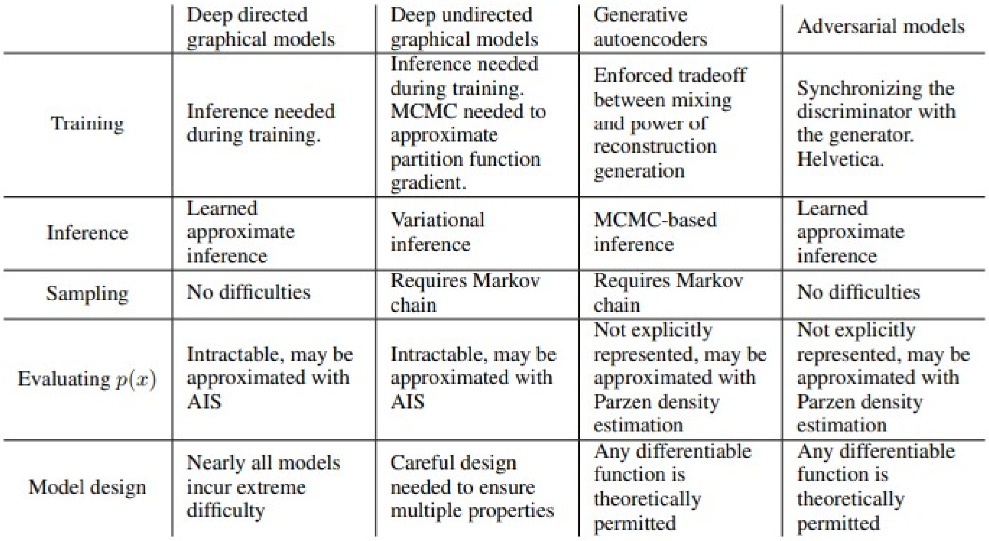
  - Hybrid model (directed+undirected)
    - Deep belief network
      - fast approximation for layer-wise training criterion
      - computation problems associated with both directed & undirected graphs.
  - Score matching and NCE
    - not targeting Log-likelihood
    - learns analytically specified PDF
    - not always possible to specify the PDF upto a normalizing constant
  - NCE
    - there is Discriminator→ not the generator – with varying distribution
    - uses fixed noise distribution (negative sampling)
  - Generative Stochastic network (GSN) → Generalized auto-encoder
    - Define a parameterized Markov chain
    - Drawing sample from desired distribution
    - not generative approach
    - requires Markov Chain sampling
    - requires feedback loops
      - Adversarial net doesn't need this, so it can use the BP
        - but have problems with unbounded activation
    - some works used BP though.
- Adversarial Net
  - Generator MLP
  - Discriminator MLP
  - Target min-max value function → Mother of GAN equation
- Theoretical analysis



Algorithm 1 Minibatch stochastic gradient descent training of generative adversarial nets. The number of steps to apply to the discriminator, k, is a hyperparameter. We used $k = 1$, the least expensive option, in our experiments.

```
for number of training iterations do
    for $k$ steps do
        • Sample minibatch of $m$ noise samples $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
        • Sample minibatch of $m$ examples $\{\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(m)}\}$ from data generating distribution $p_{\text{data}}(\boldsymbol{x})$.
        • Update the discriminator by ascending its stochastic gradient:
            $\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right)\right].$
    end for
    • Sample minibatch of $m$ noise samples $\{\boldsymbol{z}^{(1)}, \dots, \boldsymbol{z}^{(m)}\}$ from noise prior $p_g(\boldsymbol{z})$.
    • Update the generator by descending its stochastic gradient:
        $\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right).$
end for
The gradient-based updates can use any standard gradient-based learning rule. We used momentum in our experiments.
```

  - Global optimal for Discriminator
  - Convergence of algorithm 1
- Experiments
  - MNIST

  - Toronto Face Database
  - CIFAR
  - 

	Deep directed graphical models	Deep undirected graphical models	Generative autoencoders	Adversarial models
Training	Inference needed during training.	Inference needed during training. MCMC needed to approximate partition function gradient.	Enforced tradeoff between mixing and power of reconstruction generation	Synchronizing the discriminator with the generator. Helvetica.
Inference	Learned approximate inference	Variational inference	MCMC-based inference	Learned approximate inference
Sampling	No difficulties	Requires Markov chain	Requires Markov chain	No difficulties
Evaluating $p(x)$	Intractable, may be approximated with AIS	Intractable, may be approximated with AIS	Not explicitly represented, may be approximated with Parzen density estimation	Not explicitly represented, may be approximated with Parzen density estimation
Model design	Nearly all models incur extreme difficulty	Careful design needed to ensure multiple properties	Any differentiable function is theoretically permitted	Any differentiable function is theoretically permitted

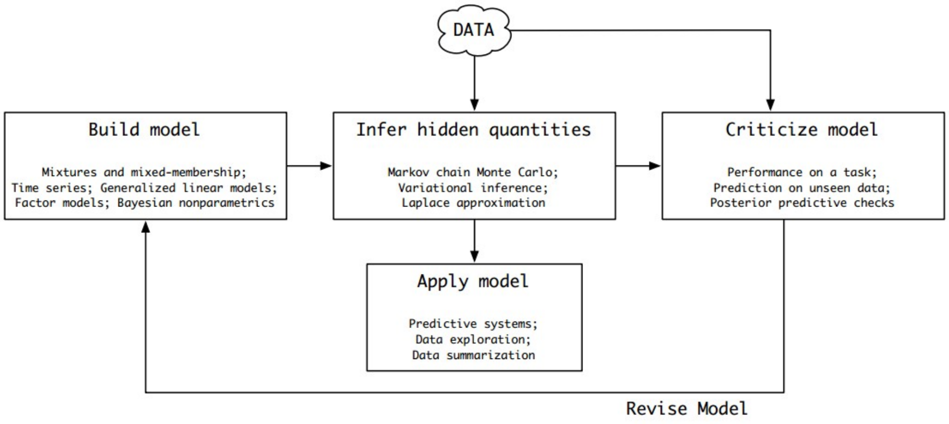
Table 2: Challenges in generative modeling: a summary of the difficulties encountered by different approaches to deep generative modeling for each of the major operations involving a model.

- Advantages and disadvantages
  - Disadvantages
    - No explicit form
    - D and G must be synchronized
      - G mustn’t trained too hard
        - Avoid Helvetica collapse or (mode collapse)
          - producing same image again and again
  - Advantage
    - Computation advantage
      - no markov chain is required → no inference during training
      - can incorporate wide varieties of function → Only need to be differentiable
    - Statistical advantages
      - learns by gradient flowing through D → no input coping opportunity
      - can learn sharp and degenerate distributions
        - Markov chain based methods learn blurry distribution
        - Degenerate: Deterministic distribution → kinda Delta function PDF
- Future works
  - Conditional Generative Models
  - Learned approximate inference
    - training an auxiliary network to predict z given x.
  - Semi-supervised learning
    - improve Discriminator performance
    - SGAN, N+1 GAN in a course project by Dr. Maryam
  - Efficiency improvement
    - accelerate in training by coordinating G and D better
      - Adaptive instance normalization
    - Taking better distribution of z

        - improving loss functions
            - Wasserstein GAN
            - Cycle consistency loss
            - Least Square GAN
            - GAN loss function (https://towardsdatascience.com/gan-objective-functions-gans-and-their-variations-ad77340bce3c)
        - Architectural modification
            - cycle gan
            - Bicycle gan
- (analysis:) https://jonathan-hui.medium.com/gan-why-it-is-so-hard-to-train-generative-advisory-networks-819a86b3750b
- More issues
    - Non-convergence
    - Mode collapse
    - Diminish gradient
    - unbalance between G and D
    - Highly sensitive to hyperparameters
- Read notes also for proof

30 Blei, D. M. (2014). Build, compute, critique, repeat: Data analysis with latent variable models. Annual Review of Statistics and Its Application, 1, 203-232.

- Survey: How to use latent variable models to solve data analysis problems
    - a probabilistic model (hidden vars encode the observable vars)
    - find posterior, condition distribution given the observation data
    - iterative processes
        - we formulate a model, use it to analyze data, assess how the analysis succeeds and fails, revise the model, and repeat.
    - Check fitness
        - predictive likelihood
        - posterior predictive checks



-
- Summary
    - probabilistic Graphical model [formulate latent variable model]
    - Mean field Variational inference [algo to approximate conditional distribution]

  - How to solve problem using latent variable model
- Latent Variable models
  - The generative probabilistic process
  - The joint distribution
  - Graphical model
  - Mixture models
    - Data are clustered and come from the distribution of the cluster.
  - Observation, hidden variable and parameters.
- Problem in explanation of equation 10!!
- Example models
  - linear factor models
  - mixed membership model
  - mixed factorization model
- Discussion
  - approximate inference algorithms that are both scalable and generic
  - develop the theory and methods of exploratory data analysis
  - Mixed-membership models posit a set of global mixture components

[Good Link](https://towardsdatascience.com/dirichlet-distribution-a82ab942a879)
[more note](http://legacydirs.umiacs.umd.edu/~xyang35/files/understanding-variational-lower.pdf)