Skoltech
Skolkovo Institute of Science and Technology

Course Project: Searching the Scientific Papers

Foundations of Software Engineering

FSE v2020.1

Alexey Artemov, Fall 2020

Outline

DISCLAIMER:
This is a game scenario, not a full real-world project

§1. Motivation

Why search through papers?

§1. Motivation

1.1. Why search through papers?

- My business: development of computer vision algorithms for geometry processing
- An average researcher needs to read *(really, skim through)* around 100-200 papers a year
  - Heavy duty when doing literature review, citing relevant work, etc.
  - Common use-case: Find a particular paper, *then* read it, *then* extract relevant information

.10453v1 [cs.CV] 20 Jul 2020

Points2Surf

Learning Implicit Surfaces from Point Clouds

Philipp Erler[1,4], Paul Guerrero[2], Stefan Ohrhallinger[1,3], Michael Wimmer[1], and Niloy J. Mitra[2,4]

[1]TU Wien [2]Adobe Research [3]VRVis [4]University College London

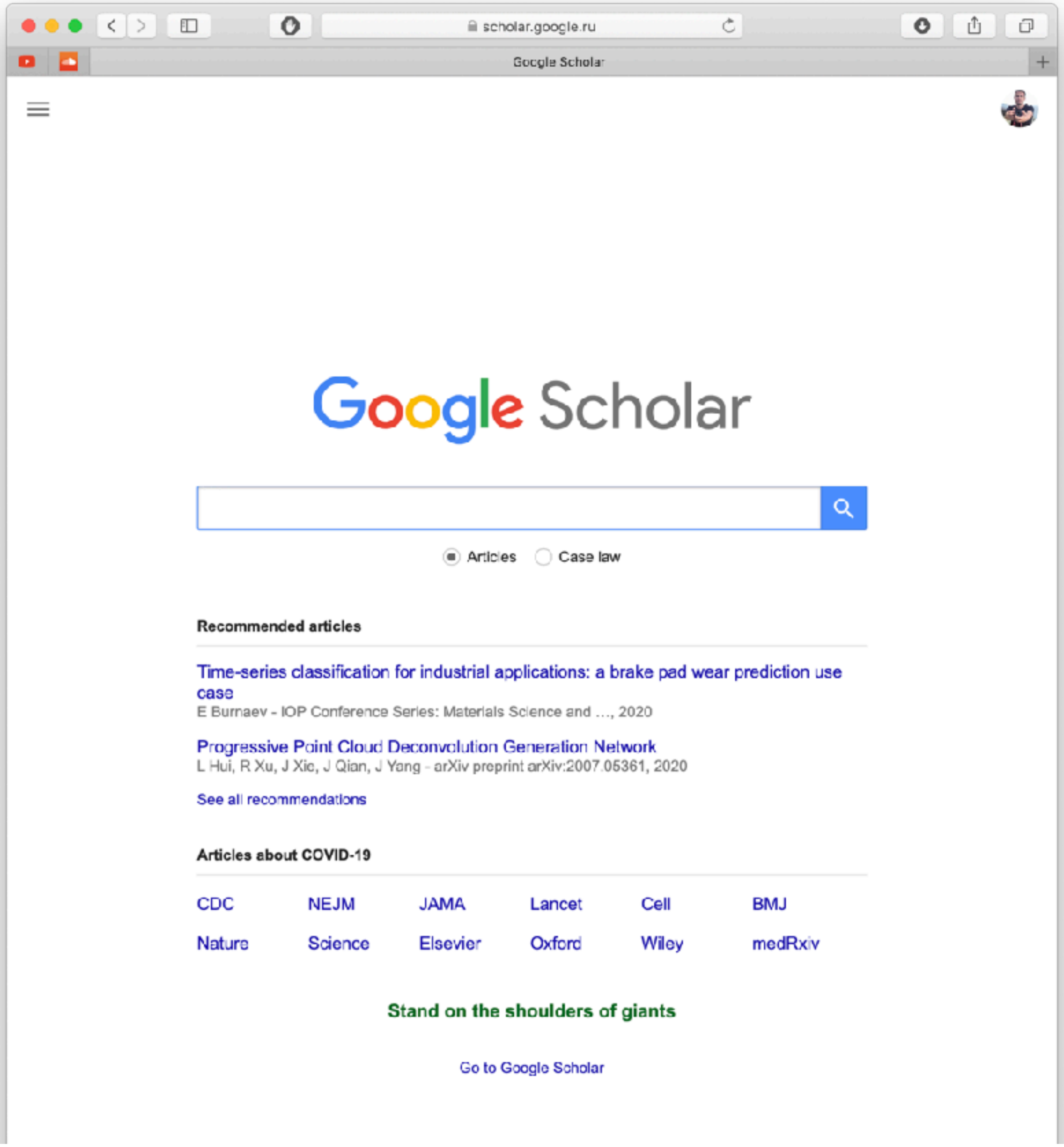
https://www.cg.tuwien.ac.at/research/publications/2020/erler-p2s/

Abstract. A key step in any scanning-based asset creation workflow is to convert unordered point clouds to a surface. Classical methods (e.g. Poisson reconstruction) start to degrade in the presence of noisy and partial scans. Hence, deep learning based methods have recently been proposed to produce complete surfaces, even from partial scans. However, such data-driven methods struggle to generalize to new shapes with large geometric and topological variations. We present Points2Surf, a novel *patch-based* learning framework that produces accurate surfaces directly from raw scans without normals. Learning a prior over a combination of detailed local patches and coarse global information improves generalization performance and reconstruction accuracy. Our extensive comparison on both synthetic and real data demonstrates a clear advantage of our method over state-of-the-art alternatives on previously unseen classes (on average, Points2Surf brings down reconstruction error by 30% over SPR and by 270%+ over deep learning based SotA methods) at the cost of longer computation times and a slight increase in small-scale topological noise in some cases. Our source code, pre-trained model, and dataset are

§1. Motivation

1.1. Why search through papers?

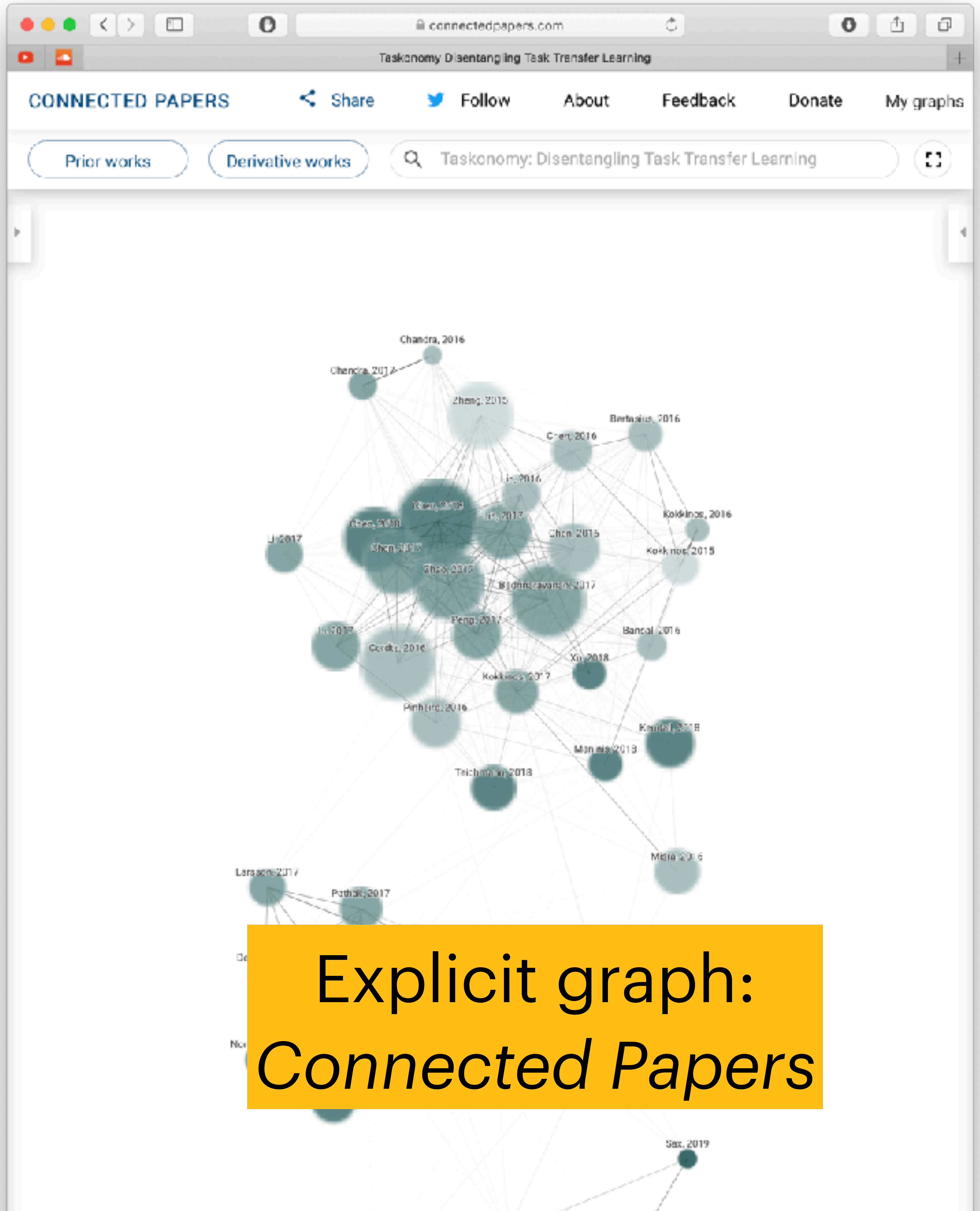
- Common ways of searching:
  - Google Scholar/Search, ConnectedPapers, Arxiv Sanity, …
- Not only the full-text search, but also the citation graph (either implicit or explicit), citation count, author search, year constraints…



§1. Motivation

Skoltech
Skolkovo Institute of Science and Technology

1.1. Why search through papers?



§1. Motivation

1.1. Why search through papers?

- In fact, one must effectively perform some form of structured reading:
  - e.g. answer questions like: does method X compare to method Y in their paper?
  - e.g. answer questions like: does method X produce output A or output B?
- This is required to perform quantitative comparison of algorithms, select relevant literature, etc.

§1. Motivation

1.1. Why search through papers?

- Example paper analysis: understanding used datasets

Datasets: We use several standard graph datasets: cora [40] (a citation network with 2,708 nodes), citeseer [40] (a citation network with 3,327 nodes), protein [14] (a protein interaction network with 3,133 nodes), adol [12] (an adolescent social network with 2,539 vertices), and fb [13, 32] (an online social network with 2,888 nodes). For facility location, we use the largest connected component of the graph (since otherwise distances may be infinite). Cora and citeseer have node features (based on

Datasets used in evaluations

- ...but some may be only cited, not used for evaluation
- ...un-cited datasets that are used in the evaluation: presented in this paper

Project goal and objectives

§1. Motivation

Skoltech
Skolkovo Institute of Science and Technology

1.2. Project goal and objectives

- **Project goal:** to obtain an extendable prototype with custom search capabilities for solving complex scientific search tasks.
- **Objective #1:** to build a baseline web-based search engine focused on full-text search over scientific papers
- **Objective #2:** to build application features based on custom search capabilities
  - (need to implement at least 1 feature from #2)

§1. Motivation

1.2. Project goal and objectives

- Objective #2: to build application features based on custom search capabilities:
  - Searching over methods that numerically compare with the given paper X
    - Example: query language, e.g. `compares:"perceptual deep depth super-resolution”`
  - Searching over methods that target solving problem X
    - Example: query language, e.g. `input:"low-resolution depth map"&input:"RGB image"` or `output:"high-resolution depth map”`
  - Searching over methods that evaluate on dataset X
    - Example: query language, e.g. `data-eval:"middlebury 2014”`
    - Example: intent recognition, e.g. `middlebury 2014` searches for methods that evaluate on dataset “middlebury 2014”

§1. Motivation

1.2. Project goal and objectives

- More requirements for the project:
  - The database with papers must be **extensible**, so that more papers may be added to be searched over later, without any involvement of the development team
  - The user interface must be accessible from the browser, be simple to use, allow downloading papers
  - The web search must be fast, e.g. respond in under 10 seconds

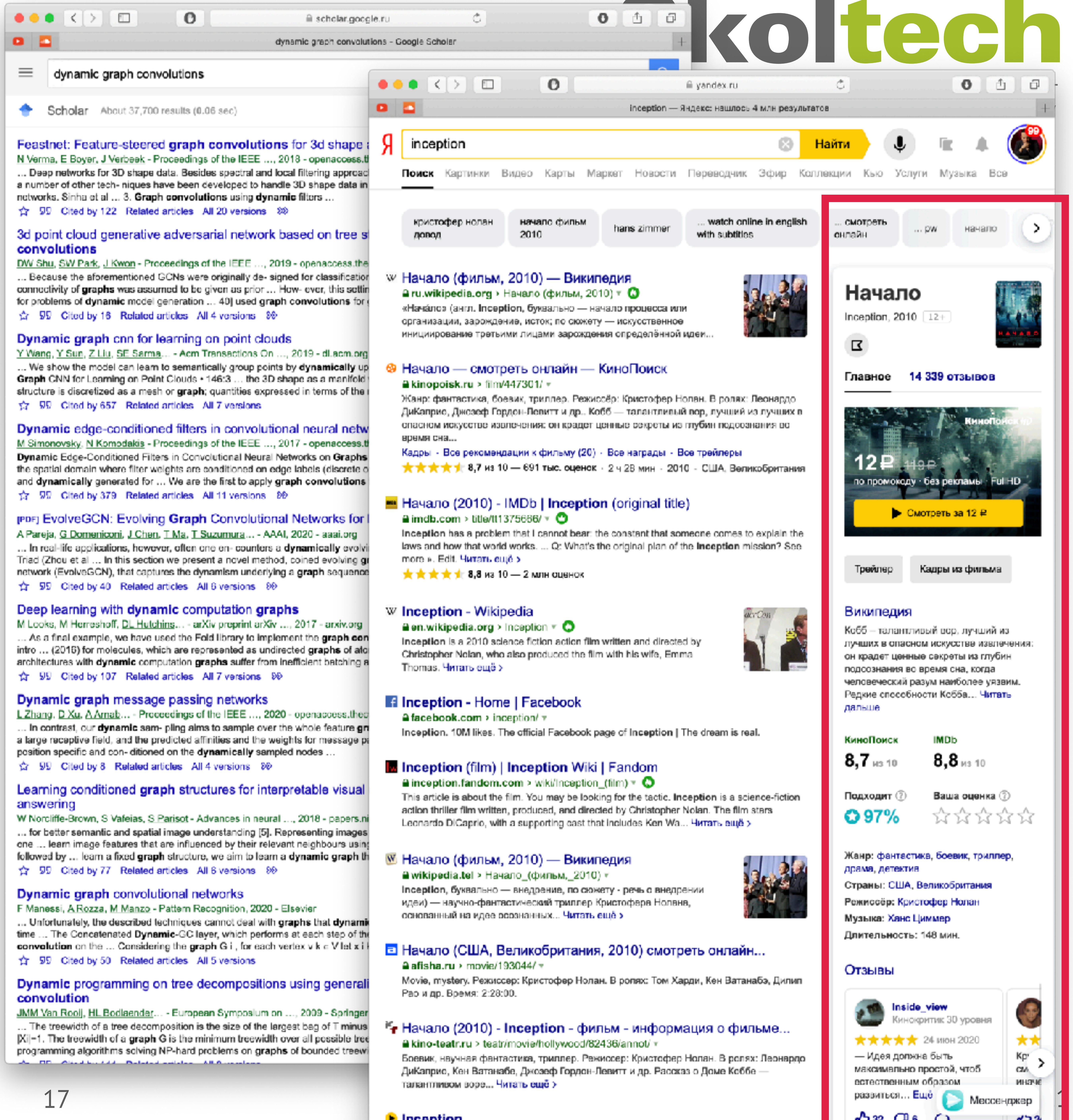
§2. Vision of a solution

Near-term prototype

§2. Vision of a solution

2.1. Near-term prototype

- Of course we only have the vision of how this looks to the users
- Might look like a standard web search, with search box and search results (GScholar)
- Might integrate custom search results like these custom snippets
- No real preference on visual appearance
- Test on e.g. https://proceedings.neurips.cc/paper/2019



Skoltech

§2. Vision of a solution

Skoltech
Skolkovo Institute of Science and Technology

2.1. Near-term prototype

- Success criteria:
  - ?
  - A "*working*" search engine? Hard to define *working* when it comes to search
  - A reasonably working search engine (meaning one has to perform some form of QA)
  - Other metrics like user satisfaction

Long-term developments

§2. Vision of a solution

2.2. Long-term developments

- Further work: mining paper argumentations
  - What does X say about Y?
  - Where does X say it improves over Y?
  - How does X differentiate itself from Y?

§2. Vision of a solution

2.2. Long-term developments

- Example argumentative writing:

> serves multi-resolution features well. Ohrhallinger et al. propose a combinatorial method [27] which compares favorably with previous methods such as Wrap [11], TightCocone [9] and Shrink [5] especially for sparse sampling and thin structures. However, these methods are not designed to process noisy point clouds. Another

Points2Surf: Learning Implicit Surfaces from Point Clouds