README: A Literature Survey Assistant

Raghav Goyal, Shih-Han Chou, and Siddhesh Khandelwal

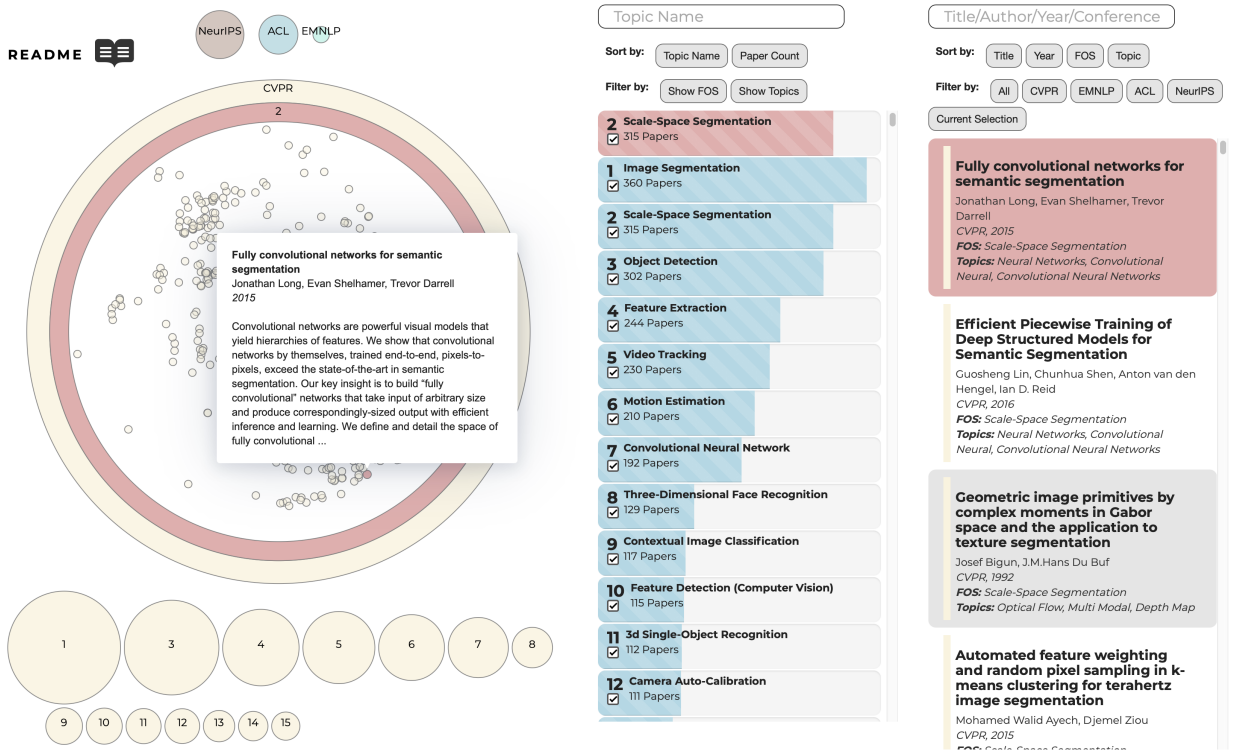

Fig. 1: Overview of our proposed literature survey tool, README. The left panel shows a hierarchical view of papers categorized based on venue and sub-fields/topics. The middle panel highlights the various automatically inferred sub-fields for a particular venue. The right panel provides an overview of all the papers currently visualized in the hierarchical view. Clicking any paper opens up the recommendation view, which is shown in Figure 7.

Abstract—Literature review is an integral element of academic research, enabling researchers to learn about and build on existing work. Traditionally, this involves manually going through various published articles, either through following the citations in a reference paper, or via keywords on sites like Google Scholar. This process can often be tedious, and there is a high likelihood of missing out on certain related literature, owing to the sheer volume of publications every year. In addition, analyzing the advances and progression in a field requires a holistic view, which manual iteration over papers lacks. To this end, we propose README, an interactive tool aimed to aide with literature reviews. README provides a holistic view of papers published over multiple years by automatically grouping them into clusters, where each cluster signifies a sub-field, thus allowing for a more structured exploration. In addition, given a reference paper the user is interested in, README also identifies and recommends relevant papers. Our current implementation uses papers from four machine learning conferences, but our tool can easily be extended to accommodate other domains and conferences.

1 Introduction

Literature reviews are crucial components to sustainable academic research. The knowledge of existing research provides a stable foundation to build upon, while simultaneously avoiding redundancy, facilitating improvements, and enabling meaningful contributions. Despite its importance, by far the most common way to conduct literature search is to manually sieve through large collections of papers, one at a time, to find work that is relevant. This exploration is done either via hopping through references within a paper, or using keywords to retrieve information from engines like Google Scholar. Due to the plethora of published work, this approach is often extremely time intensive and increases the possibility of missing out on relevant research.

Similar to [17, 22], we believe that effective visualization support for literature review can greatly benefit the scientific community. We argue for the need of a specialized framework that allows researchers to only focus on a few pertinent publications when deciding what to read next, while simultaneously providing a aggregated overview of the topic diversity in a particular field at a glance, saving both time and effort.

To this end, we propose README: an interactive literature review tool aimed at making surveying relevant research papers easier. For a particular research conference, README allows users to explore a wide range of papers published over multiple years. Additionally, publications are automatically grouped into sub-fields, thus allowing for papers targeting similar problems to be viewed at a glance. Comparison between different grouping criteria is also supported, enabling users to obtain a broader perspective of their fields of interest. The answer to the question "What should I read next?" often requires an exhaustive search

• *All authors are students at the University of British Columbia. Emails: {rgoyal14, shchou75, skhandel}@cs.ubc.ca*

through available literature. To make this is process easier, README recommends a small number of publications that might be relevant to the paper the user has expressed interest in. This is achieved using a custom similarity and coverage based relevance algorithm. Figure 1 provides an overview of the proposed framework.

This paper is structured as follows: we first list the hypotheses our work assumes. We then we describe the dataset and task abstractions, explain our relevance algorithm, and discuss the elements of our proposed visualization. Finally, we list relevant literature, and talk about the shortcomings of our framework and provide an intuition for future work.

2 HYPOTHESES

In this section we detail the hypotheses assumed to allow for an effective and efficient implementation of our proposed prototype.

H1: Each paper p can be converted into a high dimensional vector embedding $\mathbf{e}_p$, where the embedding is expressive enough to capture the semantics of the paper. That is, closeness of two papers in the resulting high-dimensional space also implies relevance of the papers to each other.

H2: Each paper p can be represented as a combination of certain base topics $t^i; i \in [1,n]$. Intuitively, one can imagine the information contained within p to be a pie. Topic t^i_p would then represent a slice of the pie.

H3: Given a query paper q, a set of candidate publications $\mathbf{P} = \{p_1, p_2, \ldots, p_k\}$ are considered "relevant" if: i) for each i, $\mathbf{e}_{p_i}$ and $\mathbf{e}_q$ are close in the high dimensional embedding space, and ii) for each topic $t^i \in q$, there is at least one paper $p_i \in \mathbf{P}$ that contains topic t^i, i.e. the set of relevant papers $\mathbf{P}$ span the space of topics in q.

H4: Each paper p can be assigned to a *single* topic t^p. Even though such a hard assignment might not be ideal in situations where p has multiple relevant topics, it allows for a tree-like hierarchical structure that is easy to parse and visualize.

The details on how these hypotheses translate to the visualization are described in the later sections.

3 DATA

There is a large selection of publications freely available online. Even though one could, in principle, scrape this data from the web, we instead use the publicly available DBLP dataset [18, 20] for feasibility.

DBLP is a *network* dataset, where each node is a paper, and a directed link between two nodes implies a citation relationship. More specifically, for two nodes A and B, a link from A to B implies paper B cites paper A. In its 11^{th} version[1], the DBLP dataset has about $4,107,340$ papers (nodes) and $36,624,464$ citation relationships (links). In addition to its size, another advantage of using this dataset is the minimal amount of data wrangling required as the dataset is already curated.

Additionally, each node has certain attributes pertaining to the paper. Some of the important attributes include

- Paper ID (Key Attribute): Each paper has a unique id that allows for easy indexing of paper details.
- Paper Venue (Categorical Attribute): The venue denotes the conference where the paper was published.
- Year (Quantitative Attribute): This denotes the publication year for each paper.
- Paper Authors (Categorical Attribute): This denotes the authors for the corresponding paper.
- Field of Study (Categorical Attribute): The domain of each paper is summarized using certain keywords, where each keyword can be thought of as an area of research.
- Field of Study Probability (Quantitative Attribute): For each field of study (FOS) within a paper, there is also an associated probability indicating the accuracy of the corresponding FOS for that particular paper.
- Citation Count (Quantitative Attribute): The number of times the corresponding paper has been sited. It should be noted that the number of outgoing links from a node is often less that the citation count.
- Paper Title and Abstract: The title and abstract for the corresponding paper. Note that the dataset does not contain the full paper content.

[1] `https://www.aminer.org/citation`

3.1 Derived Data

To make the data suitable for use in our framework, we perform certain transformations and reductions over the base DBLP Citation Network dataset [18, 20]. We now describe these in detail, linking some of them to the hypotheses mentioned in Section 2 for improved understanding.

3.1.1 Data Filtering

The 11^{th} version of DBLP Citation Network dataset contains about four million papers. It is cumbersome to perform any kind of analysis or visualization on such a large scale dataset. For feasibility, we filter papers by choosing publications from conference venues that are targeted towards machine learning and its applications. More specifically, we choose four venues: 'Computer Vision and Pattern Recognition (CVPR)', 'Neural Information Processing Systems (NeurIPS)', 'Empirical Methods in Natural Language Processing (EMNLP)', and 'Association for Computational Linguistics (ACL)'. These choices were motivated by the fact that our research interests lie in these areas, which in turn makes it easier for us to analyse the effectiveness of our framework. After filtering, we obtain a total of 28,382 papers.

3.1.2 Quantitative attribute: High Dimensional Embedding

Hypothesis **H1** assumes that each paper p can be represented as a high dimensional embedding $\mathbf{e}_p$. As the DBLP dataset does not provide the full paper content, we use the *abstract* field of each paper p as a proxy to obtain $\mathbf{e}_p$. An off-the-shelf pre-trained language model called `SciBERT` [2] is used to obtain these high dimensional embeddings. As `SciBERT` is trained over a corpus of scientific text, it is more suitable to our task when compared to other language models. Each word within the abstract of a paper p is first converted to a 768-dimensional representation using the transformer [21] module within `SciBERT`. The representations over all words within the abstract are then averaged to obtain the high dimensional embedding $\mathbf{e}_p$.

3.1.3 Quantitative attribute: 2-D coordinates

Even though the embeddings computed in Section 3.1.2 allow for comparisons in the high dimensional space, they are not feasible for the purposes of visualization. We therefore use the dimensionality reduction technique t-SNE [12] to convert the 768-dimensional embeddings into a 2-dimensional (x,y) representation.

3.1.4 Categorical attribute: Latent topics

Hypothesis **H2** presumes that each paper p can be represented using several topics t^i. The goal here is to identify a collection of words, which grouped together, constitute a particular topic in p. As each topic modeling approach has its benefits and disadvantages, we utilize two different approaches to obtain topics for each paper p. Both of these are used in our visualization, thus providing the user with a better understanding of each paper's domain. For brevity, we denote t^i as the i^{th} topic in both approaches, and correspondingly α^i_p indicates the probability of topic t^i being present in p.

Learned Topics: Using the abstracts for all papers in the reduced dataset, we leverage a probabilistic topic modeling approach called Latent Dirichlet Allocation (LDA) [4] to learn a set of 30 data-specific topics $t^i; i \in [1,30]$. Additionally, for each paper p, LDA also provides probability scores $\alpha_p = [\alpha^1_p, \ldots, \alpha^{30}_p]$ where $\alpha^i_p \in [0,1]$.

Paper id	2806265408	2560535692
Venue	neural information processing systems	computer vision and pattern recognition
Year	2018	2017
Authors	Simon S. Du, Wei Hu, Jason D. Lee	Frank Michel, Alexander Kirillov, Eric Brachmann, Alexander KrullStefan Gumhold, Bogdan Savchynskyy, Carsten Rother
Field of Study	Gradient descent, Artificial neural network, Matrix decomposition, Discretization, …	3D pose estimation, RANSAC, Conditional random field, Small number, Graphical model, …
Citation Count	1	11
Paper Title	Algorithmic Regularization in Learning Deep Homogeneous Models: Layers are Automatically Balanced	Global Hypothesis Generation for 6D Object Pose Estimation
Abstract	We study the implicit regularization imposed by gradient descent for learning multi-layer homogeneous functions including feed-forward fully connected and convolutional deep neural networks with linear, ReLU or Leaky ReLU activation. We rigorously prove that gradient flow (i.e. gradient descent with infinitesimal step size) effectively enforces the differences between squared norms across different layers to remain invariant without any explicit regularization. This result implies that if the weights are initially small, gradient flow automatically balances the magnitudes of all layers. Using a discretization argument, we analyze gradient descent with positive step size for the non-convex low-rank asymmetric matrix factorization problem …	This paper addresses the task of estimating the 6D-pose of a known 3D object from a single RGB-D image. Most modern approaches solve this task in three steps: i) compute local features, ii) generate a pool of pose-hypotheses, iii) select and refine a pose from the pool. This work focuses on the second step. While all existing approaches generate the hypotheses pool via local reasoning, e.g. RANSAC or Hough-Voting, we are the first to show that global reasoning is beneficial at this stage. In particular, we formulate a novel fully-connected Conditional Random Field (CRF) …

Fig. 2: Examples of DBLP Citation Network Dataset. For each data, it contains the paper id, venue, published year, authors, field of study, number of citations, title and abstract of the paper. Due to the space limitation, we only show a portion of the abstract.

Field of Study: The DBLP Citation Network Dataset [18] additionally, for each paper p, provides a "field of study" (FOS) value. We assume FOS to be derived from some alternate topic modeling approach, and use these FOS as topics in our framework. Each FOS value is also associated with a probability, which indicates the likelihood of that FOS being present in p.

3.1.5 Categorical attribute: Cluster Assignment

Following Section 3.1.4, although each paper p is represented using several topics t^i, following **H4**, we assign each paper to a *single* topic cluster. Such a hard assigned enforces a strict hierarchical structure, which allows for easier exploration and understanding as each paper can only be in one cluster. Each paper p is assigned to a cluster c, where,

$$c = \arg\max_i \alpha_p^i \tag{1}$$

4 Task Description

The overarching goal of this project is to build a framework that helps with the process of literature reviews. We envision the users of our tool to have access to a paper, or a list of papers, or have a notion of the sub-field they are interested in. Contrary to [17] that assume users to always have access to a seed paper, our premise is much more general. We expect users of our tool to be mainly looking for answers to two fundamental questions: i) What has already been tried in this field?, and ii) What should I read next? Our visualization, therefore, is designed to assist the user at different levels of such queries.

Understanding the existing state of research in a particular field is a key task for a researcher. The user needs to be able to do so efficiently under different scenarios. If the user has research sub-field in mind, they need to be able to explore the corresponding sub-field and analyse the papers contained within in a semantically meaningful way. In the situation where the user has a seed paper in mind, they still need to the capability to explore similar papers in the same domain.

A second important task a researcher is to figure out what papers to read next. Preferably, given a paper they are interested in, they need a curated list that has a small number of highly relevant papers. Going through this small list will save them both time and effort.

To this end, in terms of our visualization framework, we are creating a tool that allows a user to explore papers and sub-fields, query specific papers of interest for recommendations, and facilitate comparison between different recommendation criteria to provide a more holistic overview. To be more specific,

1. **Explore.** Our tool enables exploration at different query levels, ranging from a coarse conference-level query, to a research sub-field level, to an even finer query level such as a paper itself. It allows users to conduct this exploration in a semantically informed way, where relevant papers are shown at all exploration levels.

2. **Query.** Given a seed paper, our tool allows the user to obtain a set of recommended papers which are relevant to the seed paper, thus assisting in narrowing down search for what to read next.

3. **Compare.** Our tool empowers users with a choice to select and compare between different recommendation and sub-field aggregation criteria since no single criterion is *always* correct and relevant (alluding to no free lunch theorem in ML). We support two ways of identifying sub-fields: using field of study directly from the data or using latent topics extracted in an unsupervised way.

Table 1 details the analysis of README according to the What-Why-How framework [16].

5 Recommendation Algorithm

A key objective of README is to recommend a set of relevant publications, given the user has expressed interested in a particular paper. We will refer to this user selected paper as the seed paper. As described in Section 2, **H3** provides our assumed definition for relevance. Particularly, for a set of papers **P** to be recommended, we require the following two criteria to hold: i) each paper in **P** must be close to the seed paper, and ii) **P** should span the space of topics defined by the seed paper. We now provide details regarding how these two criteria are realized.

5.1 Closeness to the Seed Paper

As described in Section 3.1.2, we convert each paper p to a high dimensional embedding $\mathbf{e}_p$. The advantage of such a transformation, assuming it is expressive enough to capture the semantics of each paper,

System	*README*
What: Data	Network Data - Papers as Nodes - Citation relationship as Links
What: Derived	Attributes for each node - Quantitative Attribute: High Dimensional Embedding - Quantitative Attribute: 2-D node coordinates - Categorical Attribute: Topics - Categorical Attribute: Cluster Assignment
Why: Tasks	Discover and Locate similar nodes (papers), Query nodes of interest to obtain suggestions for other relevant nodes, Compare between different values for categorical attributes (Topics and Cluster Assignment)
How: Encode	Express nodes and categorical attribute (Topics) using spatial position and point marks, Express nodes and categorical attribute (Topics) using vertical bar chart and list idiom
How: Facet	Juxtaposed views for better navigation and comparison
How: Manipulate	Reorder data, Navigate through different hierarchical levels, Animated transitions
How: Reduce	Filter relevant nodes (papers)
Scale	20,000+ papers across 4 venues

Table 1: Analysis of README according to the What-Why-How framework [16].

is that it allows us to mathematically define "closeness" between two papers.

Particularly, given two papers p and q, the closeness s_{pq} between them can be computed as the distance between their corresponding embeddings $\mathbf{e}_p$ and $\mathbf{e}_q$. Specifically,

$$s_{pq} = \frac{1}{||\mathbf{e}_p - \mathbf{e}_q||} \tag{2}$$

s_{pq} now serves as a quantitative attribute defining 'closeness', where a larger value indicates that p and q are 'close', and a smaller value indicates the opposite.

5.2 Topic Coverage

Currently, the most common way to search for relevant papers is through search engines like Google Scholar. For a user, the process involves coming up with a keyword that is relevant to their seed paper or field, and then going through the list of results shown by the search engine. More often than not, these results include the keyword that the user provided. Such a keyword centric approach has two limitations: i) the onus of finding the right keyword that effectively summarizes the domain of a seed paper falls on to the user, and ii) papers containing a particular keyword might not cover all the aspects of the seed paper. As an example, if the seed paper encompasses topics like 'object detection', 'few-shot learning', and 'transfer learning', obtaining papers using the keyword 'object detection' might lead to an incomplete exploration of the domain.

We therefore propose a coverage based recommendation criterion to address the aforementioned issues. The objective is to suggest a handful of papers that cover the high likelihood topics for a seed paper. As described in Section 3.1.4, let $\alpha_q = [\alpha_q^1, \ldots, \alpha_q^n]$ be the topic scores computed for a seed paper q. Additionally, let $\mathbf{t}_q^\tau \subseteq [t^1, \ldots, t^n]$ be the set of topics where $\alpha_q^i > \tau$ and τ is a positive threshold. Our aim then is to find a set of publications $\mathbf{P} = \{p_1, \ldots, p_k\}$ such that for each $t^i \in \mathbf{t}_\tau^q$, there exists at least one $p_i \in \mathbf{P}$ where p_i has a high $\alpha_{p_i}^i$ value indicating that topic t^i is present in paper p_i. That is, we want to recommend a set of papers $\mathbf{P}$ where all high likelihood topics in the seed paper q are represented.

5.3 Complete Algorithm

Our complete recommendation algorithm combines the properties of the two criteria described in Sections 5.1 and 5.2. The closeness criterion ensures that the recommended papers are relevant, and the topic coverage criterion ensures that the recommended papers cover all key domain aspects of the seed paper.

In practice, the algorithm proceeds as follows,

1. Given a seed paper, we first obtain a list of 100 closest papers $\mathbf{S}$ using the criterion described in Section 5.1.
2. For the same seed paper, we filter out the top-3 most probable topics from the set of all available topics using the probability scores α_i^q. This forms the set $\mathbf{t}_\tau^q$.
3. For each topic $t^i \in \mathbf{t}_\tau^q$, we select 3 papers from the list of 100 closest papers $\mathbf{S}$, such that each of the 3 papers has a high probability of containing topic t^i. This follows from the coverage criterion described in Section 5.1.
4. Finally, these 9 papers are recommended to the user.

The choice of recommending only 9 papers is done to ensure that the user is not overwhelmed by a large number of suggestions. The user can now denote more time on going in-depth through each paper, rather than having to skim through a larger number of suggestions. If the user deems any of the recommendations useful, they can use the our algorithm again on a new seed paper to obtain more suggestions.

6 Design Solution

Our proposed tool README addresses two key objectives,

1. The ability to explore publications over the years in one place, where the publications are automatically grouped into sub-fields to enable easier search.

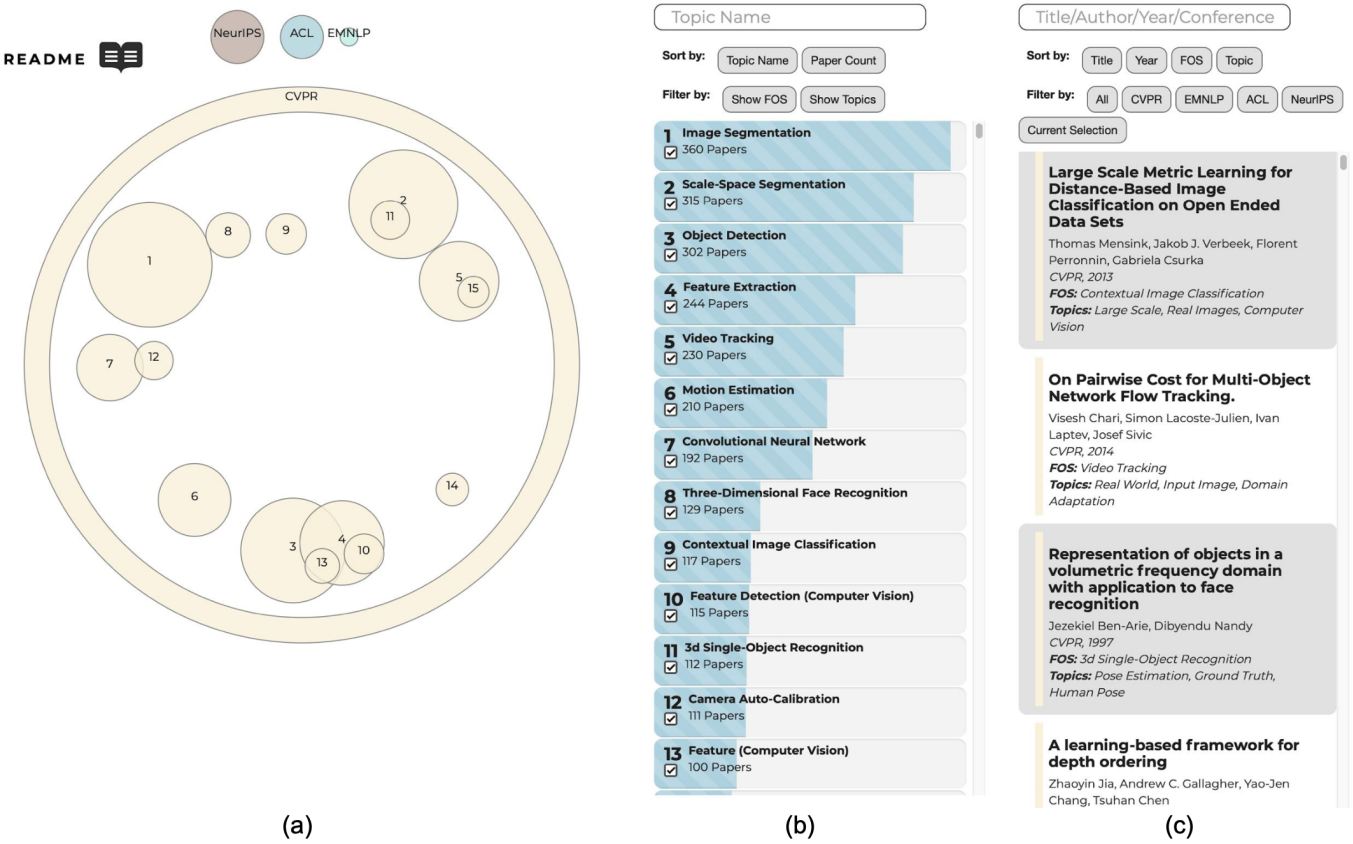

Fig. 3: **Faceted Exploration View**. (a) shows the hierarchical exploration view. Additional details on navigating this view are provided in Section 6.1.1. (b) shows the topic list view. A user has the ability to add/delete/filter the topics shown in (a). (c) shows the paper list view. A user has the ability to filter/navigate to a specific paper in (a).

2. The capability to find and suggest a handful number of relevant papers, aiding in the process of deciding what to read next.

To this end, keeping the aforementioned goals in mind, our visualization can be divided into two different views: i) the exploration view and ii) the recommendation view. In addition, our tool is interactive, thus allowing users to seamlessly navigate between various views. In the following sections, we discuss these two views in detail.

6.1 Exploration View

Contrary to existing tools such as [1] that visualize all available papers in a single view, a key decision in our design is to impose a hierarchical ordering on paper exploration. Our proposed hierarchy in shown in Figure 8. Instead of having to navigate a large collection of papers at once, as in [1], such a hierarchical ordering offers multiple advantages,

1. It reduces visual clutter as all papers are not shown at once.

2. It provides the user with additional relevant information about the domain of each paper without making them read through it.

3. It provides a broad understanding of the popularity of various sub-fields, thus providing an intuition as to where research is thriving.

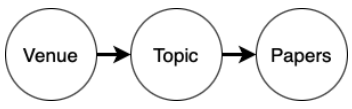

Fig. 8: Imposed hierarchy on papers.

The exploration view builds around this hierarchical ordering. As our visualization shows papers across multiple years and conferences, we make the choice to facet the data into multiple views (Figure 3). In the following sections we first describe the hierarchical view, and then discuss the linked list view.

6.1.1 Hierarchical View

Figure 4 shows the overall flow in the hierarchical view. The user initially has access to different venues (Figure 4 (a)), where each venue is denoted by a point mark. The size channel indicates the number of papers in each venue, and color is used as an additional channel to help distinguish between venues. The position channels do not encode any information, but rather just ensure that the venue marks are non-overlapping.

Clicking the point mark corresponding to a venue reveals the topic-view (Figure 4 (b)). Each topic is represented as a point mark, where the size again encodes the number of papers within the topic. The assignment of a paper to a topic follows the procedure described in Section 3.1.5. To enable better comparison between topics, the position channel is modified such that related topics are located close to each other. For each topic, the details regarding the computation of its x and y positional coordinates is further explained in Section 6.1.3. To ensure consistency and reduce cognitive load, the color channel encodes the venue.

Finally, clicking the point mark corresponding to a topic reveals the paper-view (Figure 4 (c)). We use concentric circles to highlight the granularity of exploration the user is currently at. The outer circle displays the venue, and the inner circle displays the current topic the user is navigating. Each paper is represented as a point mark. Similar to the topic-view, the position channel is intended to indicate closeness between papers. For each paper, we use the t-SNE coordinates ex-

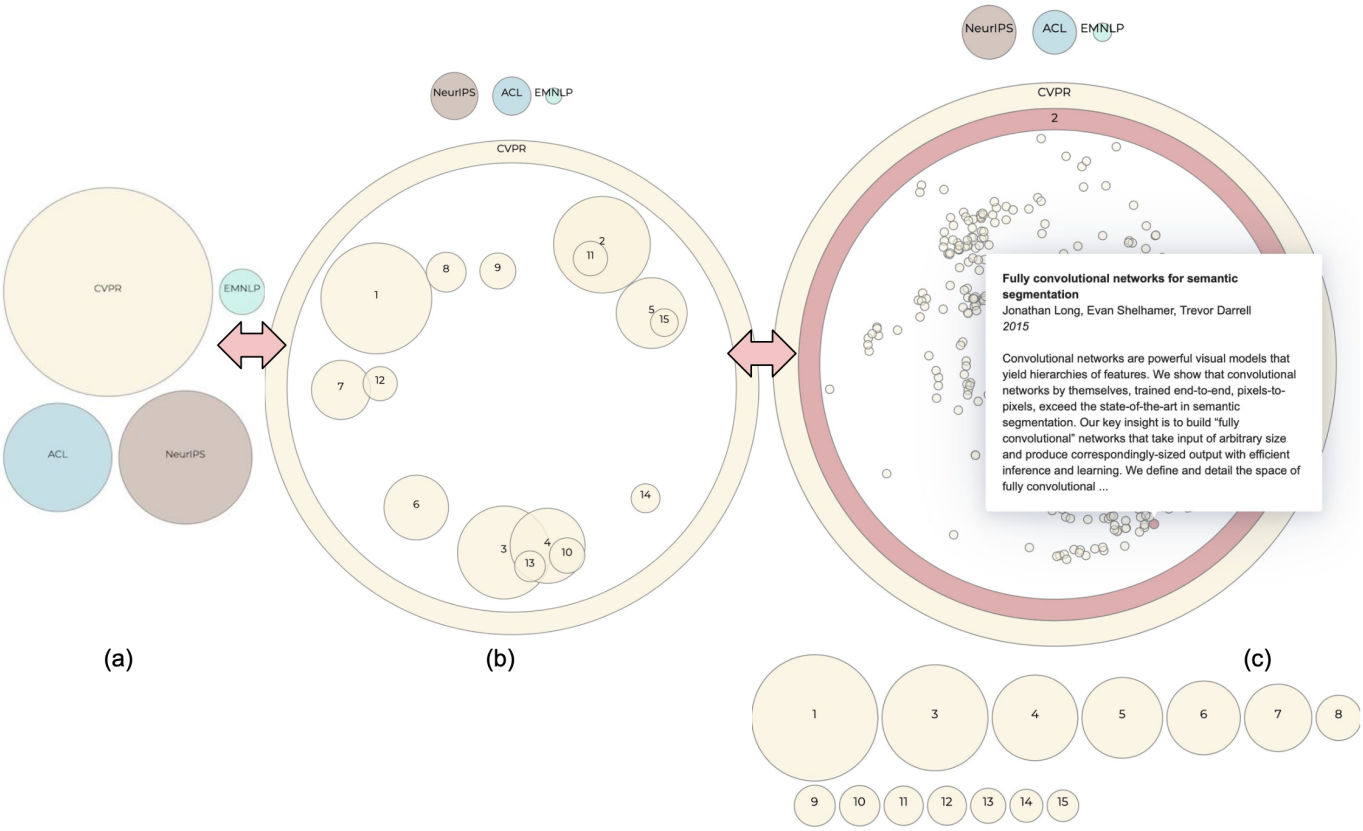

Fig. 4: **Hierarchical View**. (a) shows various venues which the users can click to reveal the topic view (b). (b) shows various topics within a particular venue where similar topics are placed nearby. Users can click on any topic in (b) to reveal all the contained papers, which can be seen in (c). Similar papers in (c) are placed nearby, and hovering on a paper reveals a tooltip with paper information. Clicking on a paper shows the recommendation view. The red arrows indicate the navigation flow.

tracted in Section 3.1.3 as the x and y positional coordinates. A tooltip provides additional details about each paper when the paper point mark is hovered over.

6.1.2 List View

As the exploration view supports $20,000+$ papers across different venues and years, having just the hierarchical view by itself would make the process of surveying different fields extremely tedious. There are several problems that contribute to this,

1. Even though the topic view (Figure 4(c)) visually provides an intuition of paper similarity and topic allocation, the user would need to individually hover over all point marks to get additional information about various papers present in the view.

2. The hierarchical view, by itself, does not allow filtering papers via keywords, or directly traversing to a paper the user is interested in.

To address the aforementioned issues, we facet the data into a two additional list views that are linked to the hierarchical view. These lists serve two distinct goals: one allows the user to interact with data on the granularity of individual papers, while the other allows interaction on the topic level. The design choice of a list idiom was additionally motivated by the improved readability it affords.

Paper List View: The paper list view is juxtaposed to the hierarchical view as shown in Figure 3(c). It allows users to directly interact with data on the level of individual papers, while simultaneously supporting searching and filtering capabilities for additional flexibility. Figure 5 highlights the different components of the paper list view, which are also explained below.

1. **Examine**: As shown in Figure 5(d), each element in the examine block corresponds to a paper, which is identified by the title, author names, venue, year of publication, and topic assignment. Additionally, the color channel is used as a strip on the left to denote the venue. By default, the examine block automatically filters papers based on the users exploration, wherein only papers accessible via the current hierarchical view (Figure 3(a)) state are shown. As an example, if the user selects a particular topic 'image segmentation' within the venue 'CVPR', the examine block will only show papers that were published in 'CVPR' and belong to the cluster 'image segmentation'. Additionally, clicking an element in the examine block alters the hierarchical view, modifying it such that the clicked paper is in view. This integration between the two views allows the user to directly traverse to the paper of choice, foregoing the need for a tedious search.

2. **Search**: Using the search bar (Figure 5(a), the user can filter papers using any keyword. The search supports partial matching, wherein the keyword can be a part of the title, author name, venue, year of publication, or the topic assigned to the paper. The search results are dynamically updated in the examine block (Figure 5(d)).

3. **Filter**: Using the filter block (Figure 5(b)), a user can quickly filter papers based on the venue, or use the 'Current Selection' option filters papers based on the current state of the hierarchical view (Figure 3(a)). In addition, the sorting options allow arranging the filtered papers in ascending or descending order, based on title, year of publication, or topic name.

4. **Paper of Interest:** When navigating the paper-view (Figure 4(c)), the hovered over paper is highlighted in red as the paper of interest

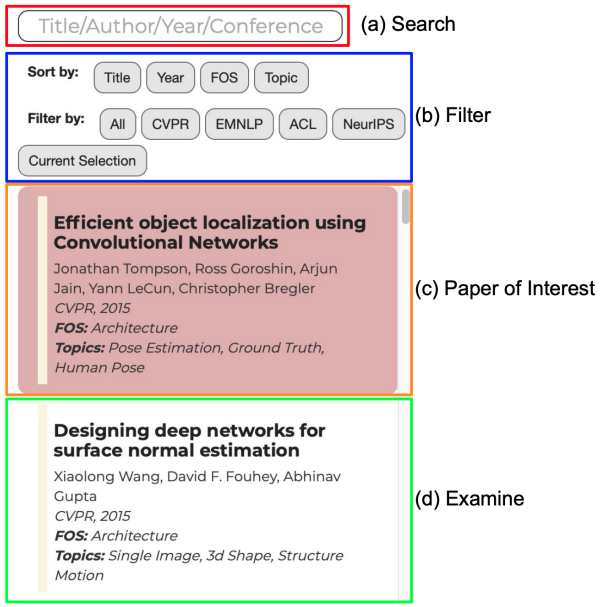

Fig. 5: **Paper List View.** (a) shows the search bar that supports partial matching based on keywords. (b) shows various quick filter options, and allows sorting the filtered papers using different criteria. (c) shows the current paper the user has selected or hovered over in the hierarchical view. (d) lists the papers obtained after various filtering or searching criteria are applied.

(Figure 5(c)). This allows a user to quickly see all the important information regarding the paper in a readable format.

Topic List View: The topic list view is juxtaposed to the hierarchical view as shown in Figure 3(b). It enables interaction with topic level data, allowing users to directly add or remove certain topics from the hierarchical view (Figure 3(a)). Figure 6 highlights the various components within the topic list view. These are further detailed below.

1. **Examine:** The examine block lists all automatically extracted topic clusters for the current venue. A vertical bar chart visually encodes the number of papers within each topic cluster. By default, the chart is sorted in descending order to highlight the more dense topic clusters. The number to the left of each list element in the chart corresponds to the point mark label in the hierarchical view (Figure 3(a)). As an example, 'Image Segmentation' would have a point mark with the label '1' in the hierarchical view. Additionally, each topic cluster is associated with a checkbox that allows the user to add or remove topic clusters from the hierarchical view . For example, if the user checks the box corresponding to 'Object Detection', an additional point mark labelled '3' would appear in the hierarchical view. Finally, clicking any element in the list opens the paper-view (Figure 4(c)) for that topic cluster.

2. **Search**: Similar to the paper list view, the search bar (Figure 6(a)) allows users to search topics using keywords. The search bar supports partial keyword matching, wherein the keyword can be a part of the topic name. The search results are dynamically updated in the examine block (Figure 6(d)).

3. **Filter**: The filter block (Figure 6(b)) enables quick sorting of topic clusters via name or count of papers within each cluster. Additionally, as mentioned in Section 4, README supports comparison between different clustering criteria. As no clustering criterion is perfect, this provides users with a more holistic understanding of the various domains in a particular research field. Our current implementation currently supports two clustering criteria: 'Field of Study (FOS)' and 'LDA Topic Modeling'. The details

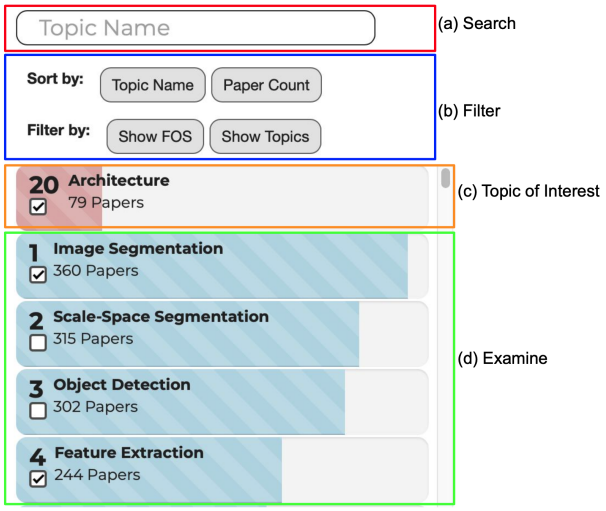

Fig. 6: **Topic List View.** (a) shows the search bar that supports partial matching based on keywords. (b) allows the user to switch between two clustering criteria, and allows sorting the filtered topics. (c) shows the topic that the user has selected or hovered over in the hierarchical view. (d) lists the topics obtained after various filtering or searching criteria are applied. A checkbox also allows the user to add or remove topics from the hierarchical view.

on how these clusters are computed are provided in Section 3.1. The 'Show FOS' makes field of study topic clusters visible in the hierarchical view (Figure 3(a)), and additionally updates the examine block (Figure 6(d)) to reflect the same. The 'Show Topics' button is analogous in behaviour, but for LDA topic modeling clusters.

4. **Topic of Interest**: When navigating the topic-view (Figure 4(b)), the hovered over topic cluster is highlighted in red as the topic of interest (Figure 6(c)). This provides additional information about the topic name and number of papers contained within.

6.1.3 Positional Coordinates for Topic Clusters

The topic view (Figure 4(b)) visually encodes topic clusters as point marks. As mentioned in Section 6.1.1, the position channel is intended to encode the relative closeness between topics. We now describe how the x and y positional coordinates for each topic cluster are computed.

For a particular topic t^i, let $\mathbf{C}^i$ be the set of papers assigned to the cluster t^i. Note that this cluster assignment is done according to the criterion described in Section 3.1.5. The positional coordinates for t^i is then defined as,

$$x = \frac{1}{|\mathbf{C}^i|} \sum_{p \in \mathbf{C}^i} p_x; \quad y = \frac{1}{|\mathbf{C}^i|} \sum_{p \in \mathbf{C}^i} p_y \tag{3}$$

where p_x and p_y are the x and y t-SNE [6] coordinates for a paper in the set $\mathbf{C}^i$. The details on the computation of these t-SNE coordinates are described in Section 3.1.3.

6.1.4 Reducing Point Collisions

In the hierarchical view, both the topic view (Figure 4(b)) and paper view (Figure 4(c)) use the position channel to encode relative closeness between entities. For topics, the computation for the positional coordinates are described in Section 6.1.3. For papers, the t-SNE [6] coordinates, as described in Section 3.1.3, are used as the positional coordinates. Naively using these coordinates, however, leads to a high number of collisions between the point marks, thus making interpretation and reasoning much harder.

To this end, we incorporate certain heuristics to reduce the number of collisions.

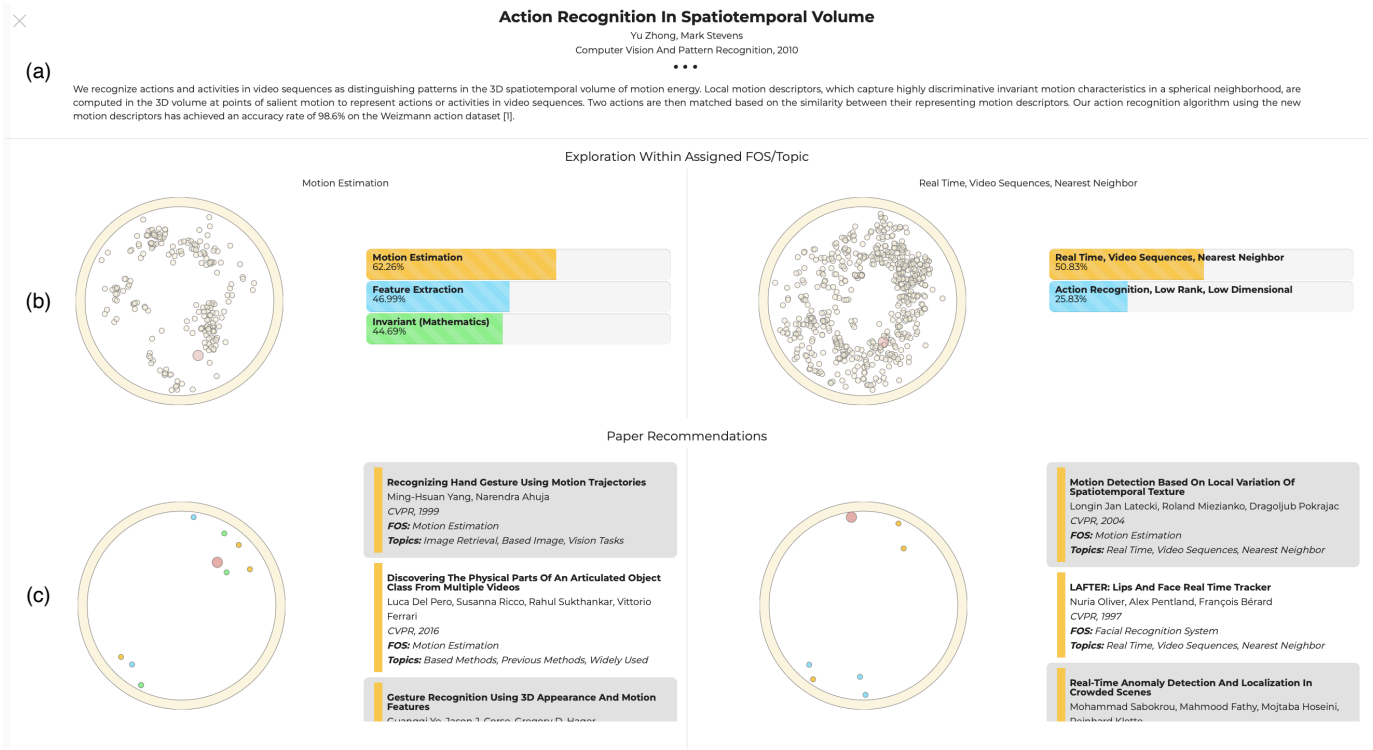

Fig. 7: **Recommendation View.** Pop-up window for paper details and recommendations. (a) shows additional details of the selected paper. (b) shows paper exploration within assigned topic and allows comparison between two different clustering criteria. Users can hover over or click on the point marks to check similar papers within the cluster. (c) shows three papers recommendations for each topic relevant to the selected paper. The colour channel is used to reduce cognitive overhead and help to associate the papers to their corresponding topic clusters.

1. **Random Jitter:** For each topic or paper, we add a random jitter to its positional coordinates. The amount of jitter is kept sufficiently small as to not induce any false trends.

2. **Radial Scaling:** The topic cluster and papers are visualized within a circular idiom. To further reduce collisions, we apply a radial scaling on the point marks to push them closer towards the circle boundary. This heuristic is motivated by observation that in a circular idiom of radius r, moving two points a and b radially outwards from the center increases the distance between them. This property is illustrated in Figure 9. The heuristic proceeds as follows,

   (a) All the points in the current view are translated such that the mean of their positional coordinates is $(0,0)$.

   (b) The circular idiom is divided into four quadrants, as shown in Figure 9.

   (c) Within each quadrant $q \in [1,4]$, we find the point a_q that is the farthest from the center $(0,0)$. The point a_q decides the amount of radial scaling that will be applied to the quadrant q.

   (d) If d_q is the distance of the point a_q from the center $(0,0)$, the new coordinates (p'_x, p'_y) for the point (p_x, p_y) located in quadrant q is computed as

$$p'_x = p_x + \frac{p_x \cdot f_q}{\sqrt{p_x^2 + p_y^2}}; \quad p'_y = p_y + \frac{p_y \cdot f_q}{\sqrt{p_x^2 + p_y^2}} \tag{4}$$

   where $f_q = r - d_q$, where r is the radius of the circle idiom.

One concern with using such a scaling is that it might remove correlations between entities that are closer to the center. However, we did not observe this phenomenon in our dataset.

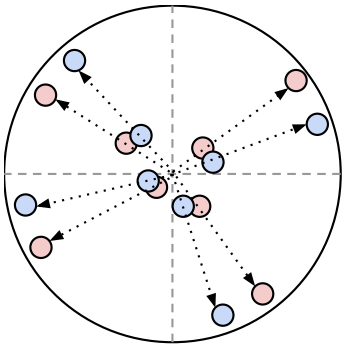

Fig. 9: **Radial Scaling**. Moving the red and blue points radially outwards increases the distance between them, therefore decreasing overlap. To ensure maximum reduction in collisions, the circular idiom is divided into four quadrants, and radial scaling is applied separately to points within each quadrant.

6.2 Recommendation View

Complementary to the exploration view, the recommendation view enables a user to delve into, in greater detail, a paper they might have interest in. This paper can either be selected through exploration, or through prior knowledge. The exploration view, which can handle both these scenarios through faceted views, can direct the user to their paper of choice.

Clicking on the corresponding point mark in the paper view (Figure 4(c)) reveals a pop-up window showing the recommendation view for the selected paper. This view is shown in Figure 7. The paper content details, including abstract, are shown to provide additional information (Figure 7(a)). For the cluster the selected paper is assigned

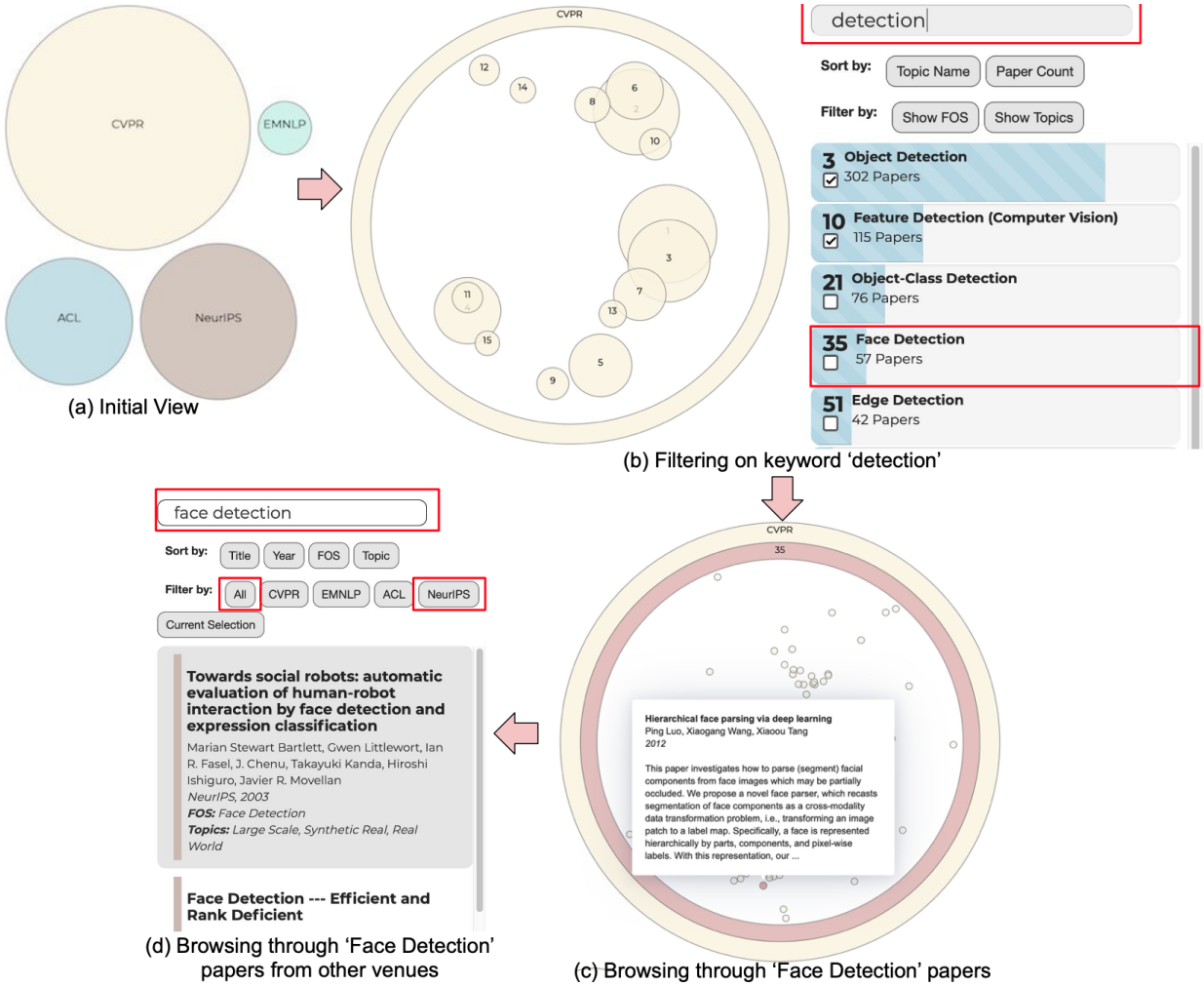

Fig. 10: **Explore: a user case study**. Please refer to Section 7.1 for additional details. A user with the intention of browsing through the vis first (a) makes a choice of a conference, which then takes the user to a faceted view (b) of research sub-fields present in the conference. The user can then search (b) for a sub-field in the topic list view. Clicking on any topic will show the semantically related papers (c). Finally, the user can explore similar papers from other conferences using the search and filter functionalities of the paper list view (d), enabling them to make an informed decision for exploring further.

to, the recommendation view also allows for a side-by-side comparison between the two clustering criteria (Figure 7(b)). For each criterion, all the papers assigned to the selected paper's cluster are visualized using point marks. A tooltip reveals additional details about each paper on hover, and the user can jump to the recommendation view for any paper by clicking its corresponding point mark. The choice of a juxtaposed view is made to facilitate easy comparison. The left side shows clusters obtained using the 'field of study' criterion. The right side similarly denotes clusters for the 'LDA Topic Modeling' criterion. Even though we make the design choice of using a hard cluster assignment to force a hierarchical ordering (Section 3.1.5), a vertical bar chart is additionally shown to provide the user with some intuition of the selected paper's topic distribution (Figure 7(b)).

Finally, for each clustering criterion, we suggest a small number of papers using the recommendation algorithm described in Section 5. For each criterion, the recommendations are faceted into two views (Figure 7(c)).

1. On the left, a point mark visually encodes each paper. The position channel indicates relative closeness, and the color channel encodes cluster name. The user can open the recommendation view for any of these papers by clicking its corresponding point mark.
2. On the right, a list provides an easy-to-read view of all the recommendations. The color of the strip visually encodes the cluster name. Similar to the other facet, the user can open the recommendation view for any of these papers by clicking its corresponding list element.

The juxtaposed view again allows for easy comparison between the recommendations obtained from different clustering criteria.

6.3 Alternate Design Solutions

We considered some alternate design solutions to before converging on the one described in Section 6. Here we detail other ideas, and discuss possible reasons as to why they are unsuitable for our problem.

Scatterplot With Link Marks: As the dataset is in the form of a citation network, our initial thought was to visualize all the papers using a point mark, akin to [1]. An additional interactive interface that would then allow the user to select a particular paper, which would in turn superimpose line marks linking the selected paper to its recommendations. Similar to Section 3.1.2, this would involve deriving a high dimensional representation from each paper, and then reducing this dense representation to a two dimensional feature space. These 2D points could then be visualized as a scatterplot. The immediate downside of this naive approach was the visual clutter, even with a subset of the total number of papers ($\sim$ 5000).

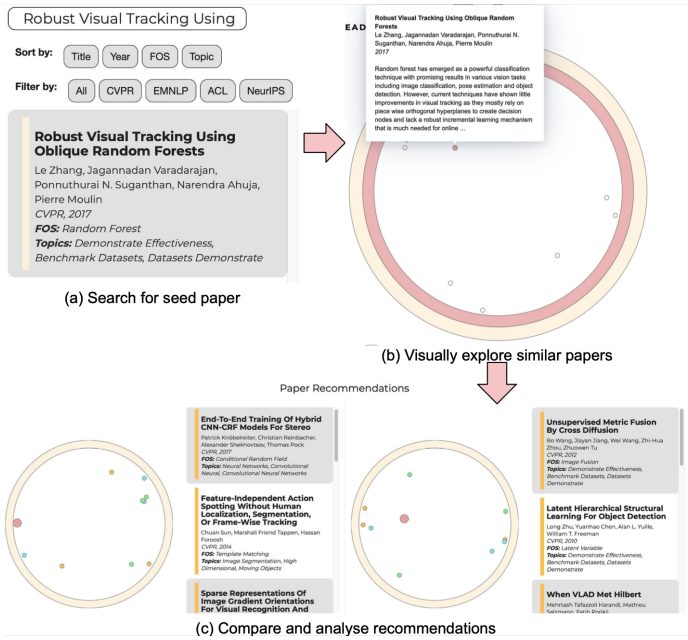

Fig. 11: **Suggest: a user case study**. Please refer to Section 7.2 for additional details. A user with the goal of getting recommendations for a seed paper can directly search for that paper in the paper list view (a). Clicking on the search result will direct the user to a view showing the semantically related papers to the seed paper (b). Finally, clicking on the seed paper point mark will show some relevant recommendations (c), which the user can explore further.

Graph with Link Marks: To solve the aforementioned issue, our next idea was to require the user to provide a query or seed paper. This query paper would be the basis for the recommendation, thus removing cognitive overload when looking at all papers in a scatterplot. Borrowing inspiration from [14], the query paper would be represented as a point mark, with links going in the upwards and downwards direction, connecting the query paper to suggested papers. This process would be recursively applied, where the suggested papers would be further connected to their suggestions. The Y-axis would encode time, where the most recent paper would be on the top. As a seed paper is required, this design limits the ability of a user to explore a particular field. We believe that, more often than not, users have a good sense of the research fields they are interested in. Providing them the option to go through different papers in this selected field provides them with a holistic view of the research diversity, thus aiding in the process of paper selection. In addition, the recursive graph generation could lead to visual cluttering as the size of the graph increases exponentially with each level.

Our proposed solution tries to address the issues of the two alternate solutions. The faceted exploration view (Figure 3) allows the user to view broad topic clusters and traverse through them without being overloaded with information. The recommender view (Figure 7) suggests relevant papers based on a selected query (which could come from exploration or prior user preference), enabling comparison between different criteria, thus further aiding the user in research.

7 Case Study

We now present a couple of scenarios that highlight two important aspects of our workflow. The first scenario describes how a user, with no specific paper in mind, can navigate through our system to obtain better overview of their research field. The second scenario talks about a user that already knows the paper they are interested in, and just want to obtain some suggestions on what to read next.

7.1 Scenario 1

John is a PhD student in the field of Computer Vision. He works in the field of detection, and wants some ideas for his next project. Therefore, he wants to look at existing work in the field of detection to get a better sense of where the research community is currently at. He uses README to help him with this task. Being familiar with the CVPR conference, he decides to explore it first. Clicking on the CVPR point mark shows him the topic view, where he is presented with some of the popular topics in the exploration view. As he is only interested in detection, he uses the search tool bar in the topic list, and is presented with a filtered list. Looking at the vertical bar chart, he easily notices that there is already a whole bunch of work in the areas of 'object segmentation' and 'feature detection'. However, the area of 'face detection' is relatively unexplored. He then clicks on the list element corresponding to 'face detection' and is presented with an overview of all the papers contained within. He can visually see sub-clusters and certain outlier by looking at the positional encoding of the paper point marks.

After browsing through some of these papers, he realizes there are some decent avenues for exploration in the field of 'face detection'. To get a complete overview, he then wants to look at papers on 'face detection' across multiple machine learning conference. He simply goes to the paper list, clicks on the 'Filter by: All' button and searches using the keyword 'face detection'. He is presented with a filtered list of all the papers that have face detection as their assigned topic in a easy to read format, which he then filters by the conference 'NeurIPS' to look at all the published papers related to 'face detection'. This workflow is illustrated in Figure 10.

7.2 Scenario 2

Susan is a machine learning PhD student working with random forest models. She has read through the paper titled 'Robust Visual Tracking Using Oblique Random Forests', and wants to get suggestions on what to read next. She uses README to help her with this task. Instead of using the exploration view, she directly searches for the paper she is interested in using the paper list view. Clicking on the paper in the filter list navigates her to the target paper. She can now visually look at all the papers that are categorized under the topic 'random forest'. But as she wants some recommendations for her seed paper, she clicks on its corresponding point mark, revealing the recommendation view. In this view, she can easily compare between different recommendations, and can navigate to any of the suggested papers by just clicking on the corresponding element. She can iterate through this process any desired number of times. This workflow is illustrated in Figure 11. For clarity, we additionally list all the recommendations for the seed paper in Table 2. It can be seen that, in addition to 'random forest', the seed paper Susan selected also delved into ideas from topics like 'convolutional neural networks' and 'pose'. To provide a more complete view and help Susan in making a more informed decision, our algorithm additionally also provides some suggestions from these topics.

8 Related Work

Existing works in literature review have used vis to show citation relationships among papers [13, 19]. Such works aim to provide an information-dense view by avoiding node-link layouts. Citeology [13] chooses to explicitly encode generations of citations in a tree-based view. CiteVis [19] uses IEEE InfoVis conference papers to draw up an year-based breakdown of papers cited by a selected paper and its citations together. Although such works are based explicitly on citation relations, our work considers semantic relations and emphasize on content based exploration which may not be present in citation graph of a paper. Other works incorporate use of faceted views with the aim to link information in different formats. PaperLens [10] uses a faceted view with histograms to show meaningful information and trends. FacetLens [11] extends faceted browsing by also making it easier to display trends and compare between them. In our work, we also employ faceted browsing between 2D visual coordinates of papers, list of topics and list of papers.

Paper type		Paper title
	Selected paper	Robust Visual Tracking Using Oblique Random Forests
Latent Topic 1 ['Demonstrate Effectiveness', 'Benchmark Datasets']	Recommended paper 1	Unsupervised Metric Fusion By Cross Diffusion
	Recommended paper 2	Latent Hierarchical Structural Learning For Object Detection
	Recommended paper 3	When VLAD Met Hilbert
Latent Topic 2 ['Neural Network', 'Convolutional Neural Network']	Recommended paper 1	Boosting Domain Adaptation By Discovering Latent Domains
	Recommended paper 2	LiteFlowNet: A Lightweight Convolutional Neural Network For Optical Flow Estimation
	Recommended paper 3	Single-Image Crowd Counting Via Multi-Column Convolutional Neural Network
Latent Topic 3 ['Synthetic Real', 'Image Sequences']	Recommended paper 1	On-line Semi-Supervised Multiple-instance Boosting
	Recommended paper 2	Decoupling Sparse Coding With Fusion Of Fisher Vectors And Scalable SVMs For Large-Scale Visual Recognition
	Recommended paper 3	Fast Concurrent Object Localization And Recognition
FOS Topic 1: 'Random Forest'	Recommended paper 1	End-to-End Training Of Hybrid CNN-CRF Models For Stereo
	Recommended paper 2	Feature-Independent Action Spotting Human Localization, Segmentation, Or Frame-wise Tracking
	Recommended paper 3	Sparse Recommendations Of Image Gradient Orientations For Visual Recognition And Tracking
FOS Topic 2: 'Pose'	Recommended paper 1	Volumetric 3D Tracking By Detection
	Recommended paper 2	SemiContour: A Semi-Supervised Learning Approach For Contour Detection
	Recommended paper 3	Leveraging Structure From Motion To Learn Discriminative Codebooks For Scalable Landmark Classification
FOS Topic 3: 'Robustness (Computer Science)'	Recommended paper 1	Discriminative Learning Of Visual Words For 3D Human Pose Estimation
	Recommended paper 2	Computationally Efficient Regression On A Dependency Graph For Human Pose Estimation
	Recommended paper 3	Global Hypothesis Generation For 6D Object Pose Estimation

Table 2: **Suggest: a user case study**. Recommendations for a particular paper are shown. The recommendations are made using Field of Study (FOS) present in the data and Latent topics extracted using LDA [4]. Three most probable papers in each category is displayed.

Task	Est. hours	Actual hours	Est. completed date	Actual completed date	Workload distribution
Ideating over project ideas	10	12	Sept 20	Sept 25	RG SHC SK
Project proposal pitch slides	3	3	Oct 1	Oct 1	RG SHC SK
Project proposal pitch video recording	3	2	Oct 1	Oct 1	RG
Literature survey	20	22	Oct 20	Oct 22	RG SHC SK
Project proposal report writing	10	10	Oct 23	Oct 23	RG SHC SK
Project direction finalization	4	3	Oct 24	Oct 25	RG SHC SK
Finalizing dataset	6	6	Oct 24	Oct 25	RG SHC SK
Data gather and filtering	15	17	Oct 28	Oct 30	SHC
Data pre-processing	25	30	Nov 4	Nov 7	RG
High dimensional embedding	10	12	Nov 7	Nov 10	RG SHC SK
Topic modeling	15	15	Nov 14	Nov 14	RG SK
Project updates report writing	6	8	Nov 18	Nov 18	RG SHC SK
Design of the exploration view	35	40	Dec 1	Dec 5	SK
Design of the recommendation view	25	30	Dec 5	Dec 7	SHC SK
Project final presentation slides	5	7	Dec 10	Dec 10	RG SHC SK
Project final presentation video recording	6	6	Dec 10	Dec 10	RG
Final project report writing	15	21	Dec 14	Dec 14	RG SHC SK

Table 3: **Milestones**. Milestone specifications and rough hour estimates. RG: Raghav Goyal, SHC: Shih-Han Chou, SK: Siddhesh Khandelwal.

Content-based exploration uses a semantic similarity measure to recommend papers. Citeomatic [3] uses a nearest-neighbor approach for a query and paper similarity where the query can any text. On the other hand, our approach takes query as a paper and loses some generality. ForeCite [9] employs heuristics and identifies concepts by using number of citations as an additional information, where the intuition is that a popular concept often has a paper that is disproportionately cited. Our approach takes a simpler route and relies on semantics extracted by a pretrained language model [2]. PaperQuest [17] proposes a multi-level filtering of relevant papers together with user's interest and preference. However, in this work we consider a single-level decision based on user's query only.

From visualization perspective, a 2D scatter plot of papers formed using a dimensionality reduction technique such as t-SNE [12] or UMAP [15] is not new and has been used extensively to visualize related papers where closeness in 2D space represents similarity [1, 7]. ICLR Paper Explorer [1] uses an interactive scatter plot where hovering over a paper point mark produces its title, author and a representative figure. Adjutant [7] goes a step further to form topic clusters from related papers in an unsupervised fashion that allows for topic-based exploration. In this work, we also support topic-based exploration, however we have two criterion for topic extraction: one taken from metadata (Field of Study) and the other extracted in an unsupervised manner [4], with an aim to increase coverage. Recently, Covid-19 pandemic has sparked a lot of research work making the need of a literature review system ever more crucial. SciSight [8] creates a user query based graph of authorship connections. A node or 'card' in the graph denote a research area, authors associated with the area and their affiliations. This lets users to make sense of who is working on what. In our work, we represent clusters as the centroid of the papers within them, however inter-cluster relations is something that can be explored as future work.

9 Implementation Details

The implemented framework has two main components, the first being the preprocessing and deriving additional data, and the second being the front-end visualization. We discuss the implementation details of these two components in detail.

9.1 Preprocessing and Derived data

We use Python language[2] for this part of our work. We generate data for each conference separately which contains papers and their references in a single file. For each paper, we extract features based on a pretrained language model SciBERT[3] by feeding in abstracts to the model. We then reduce the dimensions of the resulting features by using t-SNE[4] and choosing *perplexity* and *learning rate* to be 20 and 100 respectively. We use GPU-accelerated version of t-SNE and feature extraction pipeline so as to have a reasonable runtime for thousands of papers. For topic discovery, we use public implementation of LDA [4] from scikit-

[2] `https://www.python.org`
[3] `https://github.com/allenai/scibert`
[4] `https://github.com/CannyLab/tsne-cuda`

learn[5]. We build a corpus out of abstracts from papers with vocabulary of top 500 terms with bigrams and trigrams only since single words did not give meaningful topics. Also we use standard stop words lists to filter out some of commonly occurring words in English, and extended it to our case to include words such as `['paper', 'presents', 'propose', 'approach', 'outperform']`. We extract 30 topics for each conference. For recommendation part, we select 100 nearest-neighbours of each paper based on euclidean distance, and make recommendations for top-3 topics based on field of study and latent topics. All this data is then parsed to json format for visualizing.

9.2 Frontend Application

The frontend application is implemented as a webpage. Most of the functionality in our visualization interface is implemented from scratch on top of D3.js library [5], with the exception of the topic and paper list view. These list views, including the search and filter functionalities, are implemented on top of the list.js library [6]. All animations and interactions are implemented using a combination of D3.js, Javascript, and jQuery [7]. Their implementation is based on listening to browser events (like click and mouse-over) and updating the application state accordingly. The entire visualization interface is shown on a single page, and any changes are shown through smooth transitions. The recommendation view is implemented as a pop-up window to prevent the user from having to navigate multiple pages.

Instead of implementing a database server, for simplicity and given the amount of time we had, the frontend application directly uses the pre-processed json format files for the purposes of visualization. Making the use of a sophisticated server that can be queried is one of the future directions to explore.

10 Discussion and Future Work

README aims to aid the process of literature reviews. Based on our personal experiences, with the exponential growth in the number of publications in fields like machine learning over the recent years, it is becoming increasingly difficult to keep track of all relevant research. The motivation behind README was to put forth a framework allows users to effectively explore through a large number of papers, while simultaneously helping them answer the tricky question of what to read next.

Contrary to existing tools like [1], one of the main strengths of our work is the imposition of hierarchical ordering on exploration. We believe that it is easier for users to categorize research in terms of sub-fields and venues. Additionally, we think researchers would prefer only seeing research from their areas of interest rather than papers from all fields, which is the case in [1] and also in various conference proceedings.

We additionally feel that paper recommendation systems, in addition to relevance, should also focus on the 'coverage' criterion. That is, suggestions to the user should take into account all the aspects of a particular paper, and not just use metadata or citation relationships. To that end, our proposed recommendation algorithm is a step in this direction. Though the algorithm isn't perfect, we believe this is a good starting point for future exploration.

Our work is not without its own set of limitations. For one, it still does not scale well to millions of papers, as even the hierarchical view gets too crowded with more number of papers being assigned to a single topic. In our current implementation, the recommendations for a particular paper only come from the same venue. We envision scaling this to incorporate different venues in the future. Additionally, our recommendation algorithm might miss some relevant papers as it relies on custom closeness and coverage heuristics. Even though we expect it to perform reasonably well, more experimentation and analysis is required to have a better understanding. Finally, using t-SNE [6] to obtain the positional coordinates for papers and topic clusters might induce incorrect correlations, leading to users inferring inaccurate trends within the data.

For future work, one of our priorities is to facilitate trend analysis through a dedicated visualization. More specifically, we want to aide the understanding of the evolution of a field over time, and simultaneously look at the impact of a particular paper on the field itself. We feel this additional information would greatly help users in their paper selection process. We also want to improve upon our recommendation algorithm, particularly by incorporating citation count into the algorithm formulation. Intuitively, highly cited papers have a greater influence on the research community, and therefore should be preferred when suggesting what to read next. Additionally, to improve flexibility, we also want to give the user control over which criteria to use in the recommendation algorithm based on their needs. Finally, our implementation, in its current state, is not deployable. This can be attributed to the use of locally pre-processed files and non-responsive web interface. In the future, we would like to integrate a database that can be queried on the fly, and incorporate support for different screen resolutions.

11 Conclusion

We propose README, an interactive visualization aimed at helping users with the process of literature reviews. Our tool allows users to explore a wide range of papers, taken from different venues and ranging across multiple years, all in one place. Our design choice of a hierarchical ordering helps with reducing visual clutter, while simultaneously enabling users to look and compare between papers within a topic at a glance. Faceted views with different search and filtering options makes going through a large number of papers much easier. Finally, for a particular paper of interest, we suggest what to read next by recommending a handful of papers generated using a custom coverage and relevance based algorithm.

Acknowledgments

We would like to thank professor Tamara Munzner for her continued guidance and support throughout the project.

References

[1] Iclr paper explorer. `https://iclr.cc/virtual_2020/paper_vis.html`. Accessed: 2020-10-22.

[2] I. Beltagy, K. Lo, and A. Cohan. Scibert: Pretrained language model for scientific text. In *EMNLP*, 2019.

[3] C. Bhagavatula, S. Feldman, R. Power, and W. Ammar. Content-based citation recommendation. *NAACL*, 2018.

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[5] M. Bostock, V. Ogievetsky, and J. Heer. D^3 data-driven documents. *IEEE transactions on visualization and computer graphics*, 17(12):2301–2309, 2011.

[6] M. Brehmer, S. Ingram, J. Stray, and T. Munzner. Overview: The design, adoption, and analysis of a visual document mining tool for investigative journalists. *IEEE transactions on visualization and computer graphics*, 20(12):2271–2280, 2014.

[7] A. Crisan, T. Munzner, and J. L. Gardy. Adjutant: an r-based tool to support topic discovery for systematic and literature reviews. *Bioinformatics*, 35(6):1070–1072, 2019.

[8] T. Hope, J. Portenoy, K. Vasan, J. Borchardt, E. Horvitz, D. S. Weld, M. A. Hearst, and J. West. Scisight: Combining faceted navigation and research group detection for covid-19 exploratory scientific search. *bioRxiv*, 2020.

[9] D. King, D. Downey, and D. S. Weld. High-precision extraction of emerging concepts from scientific literature. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020.

[10] B. Lee, M. Czerwinski, G. Robertson, and B. B. Bederson. Understanding research trends in conferences using paperlens. In *CHI'05 extended abstracts on Human factors in computing systems*, pp. 1969–1972, 2005.

[11] B. Lee, G. Smith, G. G. Robertson, M. Czerwinski, and D. S. Tan. Facetlens: exposing trends and relationships to support sensemaking within faceted datasets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1293–1302, 2009.

[5] `https://scikit-learn.org/stable/index.html`

[6] `https://listjs.com`

[7] `https://jquery.com`

[12] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 2008.

[13] J. Matejka, T. Grossman, and G. Fitzmaurice. Citeology: visualizing paper genealogy. In *CHI'12 extended abstracts on human factors in computing systems*, pp. 181–190. 2012.

[14] M. J. McGuffin and R. Balakrishnan. Interactive visualization of genealogical graphs. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, pp. 16–23. IEEE, 2005.

[15] L. McInnes, J. Healy, and J. Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[16] T. Munzner. *Visualization analysis and design*. CRC press, 2014.

[17] A. Ponsard and F. Escalona. Paperquest: a visualization tool to support literature review. `https://www.cs.ubc.ca/~tmm/courses/547-14/projects/antoine-pax/report.pdf`, 2014.

[18] A. Sinha, Z. Shen, Y. Song, H. Ma, D. Eide, B.-J. Hsu, and K. Wang. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th international conference on world wide web*, 2015.

[19] J. Stasko, J. Choo, Y. Han, M. Hu, H. Pileggi, R. Sadana, and C. D. Stolper. Citevis: Exploring conference paper citation data visually. *Posters of IEEE InfoVis*, 2, 2013.

[20] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008.

[21] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[22] X. Zhang, Y. Qu, C. L. Giles, and P. Song. Citesense: supporting sensemaking of research literature. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 677–680, 2008.