RI:Small: Beyond Titles and Abstracts: Text + Links are Better Together

Project Summary

Opportunity: Scientists around the world use search engines such as Google Scholar to find academic papers. There has been considerable interest recently in language models (BERT, ChatGPT) and graphs (graphical neural nets (GNNs)) for academic search and other applications in Natural Language and Information Retrieval: web search, recommendations, collaborative filtering and traffic analysis (for the intelligence community). Benchmarks tend to focus on clean data snapshots from a few years ago, but it is important to experiment with large (and growing) citation graphs to appreciate network effects (Metcalfe's Law) and timeliness. We need methods to cope with realities such as missing values and bad data. Incremental updates are needed to keep up with growth.

In this project proposal, four use cases will be considered: (1) recommendations, (2) finding experts, (3) routing submissions to reviewers and (4) summarization of sets of documents, emphasizing pairwise similarities between documents (unlike 'tl;dr' methods, which process each document one at a time, missing relations across documents). Similarity plays a key role in these use cases. There is an opportunity for creative ensembles of deep nets and spectral clustering. Deep nets have been effective for capturing similarities of texts, and spectral clustering is promising for links. Spectral clustering can be viewed as a generalization of PageRank with more than one hidden dimension (Eigenvector). For routing submissions to reviewers, many systems focus on titles and abstracts, but not references. To capture references, we need methods that take advantage of links at inference time, unlike GNNs, which use links for fine-tuning, but not for inference. Since most submissions cite recent literature, it is important to keep our data structures and benchmarks up to date. Web search companies know benchmarks need to be updated frequently because the web is a moving target. So too, timeliness is important for our use cases.

Keywords: Artificial Intelligence; Deep Nets; Graphical Neural Networks (GNNs) & Spectral Clustering; Academic Search; Summarization; Citations

Intellectual Merit: We will take advantage of multiple representations (embeddings). Some are better for matching on abstracts and others are better for matching on citations. We view the literature as a conversation between authors and the audience, somewhat like social media. Some representations are better for capturing the perspectives of the authors, and others are better for capturing perspective from the audience. The former are time invariant since papers do not change after publication, but the latter evolve as more citations flow in, and new perspectives take hold within the community.

Ensembles of diverse perspectives have many advantages. Combinations of perspectives are robust to realities such as missing values and bad data. Diversity over representations creates opportunities for diverse computing environments. We will use deep nets for the author's perspective, and spectral clustering for the response from the audience. GPUs with GigaBytes of RAM are popular for deep nets; CPUs with TeraBytes of RAM may be preferable for spectral clustering.

Broader Impacts: This proposal will make it easier for everyone to make better use of the literature. There are many practical applications such as topic classification, information retrieval, recommender systems, expertise finding and routing papers to reviewers. This will help us make optimal use of expertise, significantly improving the lives of people in resource-poor countries and environments by, for example, encouraging scientific experts to work on critical projects.

Project Description

Introduction

Scientists around the world use search engines such as Google Scholar to find academic papers. There has been considerable interest recently in language models (BERT [1]) and graphs (graphical neural nets (GNNs) [2]) for applications in Natural Language and Information Retrieval such as: web search, Academic Search [3, 4], recommendation systems, collaborative filtering and traffic analysis (for the intelligence community). We will focus on Academic Search because of the availability of data. Semantic Scholar [4] supports bulk downloads and ad hoc queries for many fields in their databases including titles, abstracts, authors and references. This data makes it possible to construct a citation graph of $N \approx 200$M papers (nodes) and $E \approx 2$B citations (edges). (We will use k for thousands, M for millions, B for billions, GB for Gigabytes and TB for Terabytes.)

Intellectual Merit: We propose ensembles of multiple representations. Deep nets such as Specter [5] have been effective with text (abstracts); spectral clustering will be more effective with links (citations). Combinations of the two are better than either by itself. Although it is tempting to combine representations into a single representation (with GNNs), there are advantages to modularity:

1. coping with realities: missing values and bad data,
2. maintenance and incremental updates,
3. diversity over computing environments, and
4. theoretical understanding

Benchmarks [6] tend to focus on clean snapshots, but missing values and bad data are a reality. Error correction is possible when we can compare across diverse/multiple perspectives.

Modularity makes it easier to support incremental updates because some embeddings vary with time, and some do not. Incremental updates are important because the citation graph is large and growing. Benchmarks tend to freeze snapshots from a few years ago, but that is unrealistic. Web search companies update their benchmarks frequently because the web is a moving target.

Modularity also has consequences for computing environments. GPUs with GigaBytes of RAM are popular for deep nets; CPUs with TeraBtyes of RAM may be preferable for spectral clustering.

Finally, modularity has consequences for theoretical understanding. Following [7], we will compare representations based on deep nets with representations based on spectral clustering and linear algebra. We hope to generalize results from relatively well-understood linear algebra to deep nets.

Multiple Representations

Our project is intended to show that multiple representations of documents are better together, as illustrated in Figure 1. This proposal will make use of both text and citations at inference time. A number of systems such as Specter and GNNs take advantage of citations when fine-tuning, but not at inference time. The goal of these fine-tuning processes is to produce a model, f, that will be applied at inference time to a string, s, typically titles and abstracts. We are concerned that this approach misses some important opportunities. Consider the task of routing submissions to reviewers. Ideally, the reviewer should be qualified and sympathetic with the basic approach. How do we find such reviewers? When we started EMNLP, we used references in submissions. Reviewers cited in the references are likely to be qualified and sympathetic. The system for routing submissions to reviewers should take advantage of references, as well as titles and abstracts.

In this work, we generalize f to take advantage of whatever information is available including titles, abstracts and references. This approach improves coverage and robustness. Estimates of similarities will succeed even when abstracts, references and other properties of documents are corrupted and/or missing at inference time. That is, we assume document ids, d, are associated with various properties that are often available (but not always):

- text: titles, abstracts, tl;dr (too long; didn't read) summaries, full text, and
- context (links): citations, citing sentences, plus
- more: authors, venues, fields of study

In this way, the proposed approach:

1. expands coverage to papers that have links, but do not have abstracts,
2. enables error detection, which is important when abstracts are corrupted (Table 3), and
3. enables queries over strings (titles and abstracts), links (citations) and more.

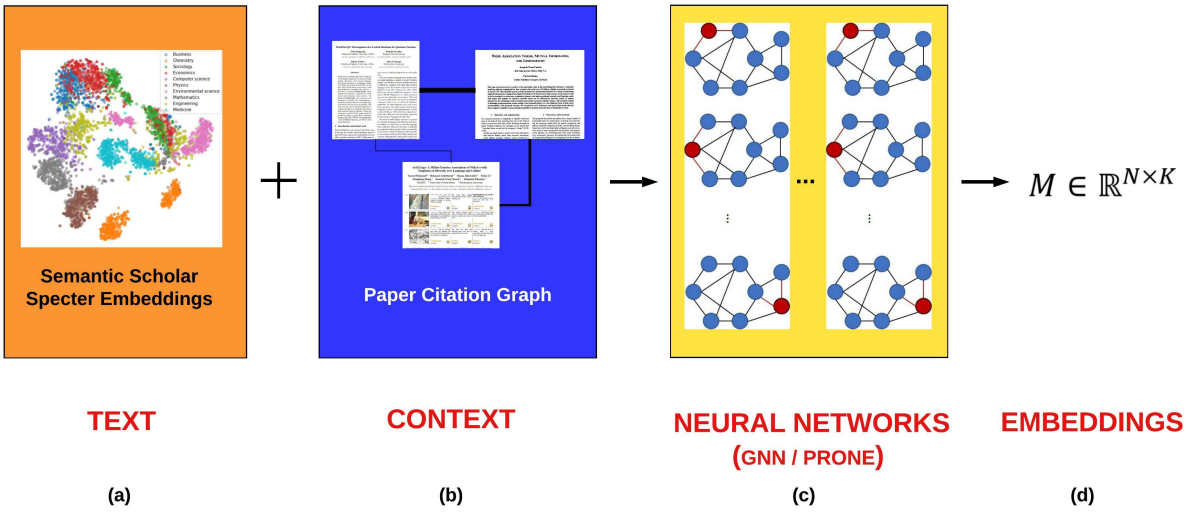
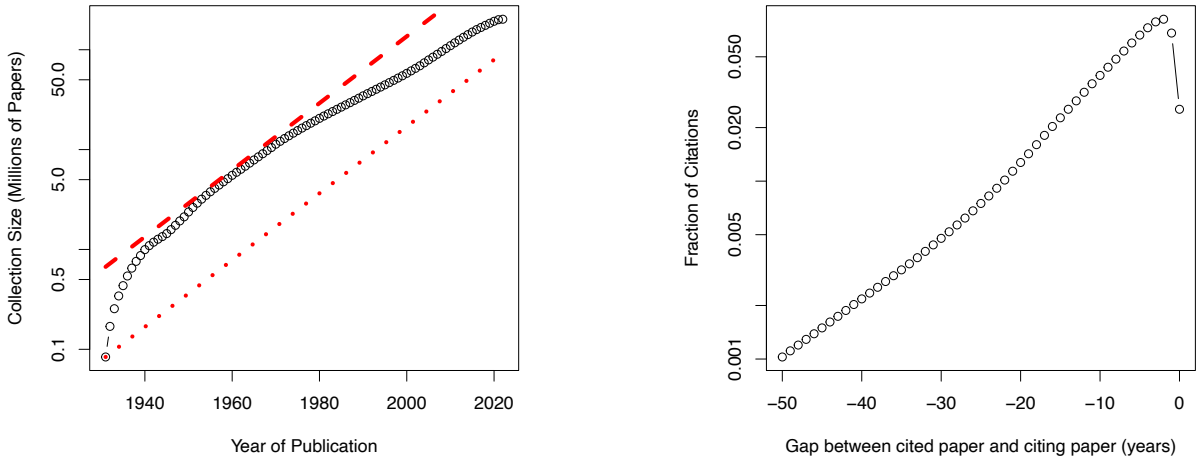


Figure 1: Embeddings are typically based on text (titles and abstracts). We will create multiple embeddings based on combinations of text and context (links such as citations and citing sentences).

Time Invariance And Incremental Updates

As mentioned above, missing values and bad data are a reality. Growth is another reality. It feels harder than it used to be to keep up with the literature. Figure 2 shows the academic literature is a moving target (like the web). According to [8], "global scientific output doubles every nine years." The observed points (black circles) fall between the red lines, confirming the 9-year prediction in [8]. The two red lines double every nine years with different initial conditions.

Scale is both a challenge and a blessing. Metcalfe's Law [9] dates back to 1979, when 3Com was selling networks with three devices, $n = 3$, two computers connected to a printer. Metcalfe argued that they should sell larger networks (and more 3Com products) because of economies of scale, since benefits scale with n^2 edges, faster than costs, which scale with n nodes. As Metcalfe argues in a survey of 40 years of Ethernet [10], his law has been good for internet companies (Google,



(a) Observations [4] (black); predictions (red).

(b) Most citations refer to recent publications.

Figure 2: Semantic Scholar [4] is large (200M papers), and doubles approximately every nine years.

LinkedIn, Facebook, Twitter, and Snapchat). We expect his law to be good for Academic Search as well.

Timeliness is important for web businesses (and Academic Search). The citation graph can be partitioned along the time dimension. In this way, we can study the evolution of the graph by rolling the graph back in time. We will use older views of the graph to predict more recent views.

It is natural to model a paper as cast in stone when it is published, but we prefer to view the literature as a conversation, somewhat like social media. For example, while Turing's paper [11] has not changed since it was published in 1936, there is a large (and growing) body of work that builds on his contribution. The value of a paper to society is a combination of the primary source plus audience appreciation (subsequent literature and secondary sources).

Some embeddings, M, evolve over time, and some do not. The value of a paper combines time-invariant contributions from authors (Specter embeddings of abstracts, M_S), with contributions from the audience (proposed embeddings, M_P, based on citation graph). Abstracts do not change after publication unlike citations that accumulate over time. Thus, M_S is time invariant, unlike M_P. In this project, we would like to study the evolution over time of embeddings and impact.

Some embeddings take a long time to compute from scratch. Given G's rapid growth (as shown in Figure 2), it is highly desirable to support incremental updates. One suggestion for incremental updates is to compute embeddings for overlapping sets. Suppose we have a set of papers published before t_1, plus a set of recent updates published after t_2. Let M_1 and M_2 be embeddings for the two sets. We then need a method to align vectors in M_1 with M_2. If we construct M_1 and M_2 with a large overlap (where $t_2 \ll t_1$), then we can compute O_1 and O_2 to be the vectors from M_1 and M_2 for the overlapping papers. One could hope to find a rotation, $R \in \mathbb{R}^{K \times K}$, with the Orthogonal Procrustes Problem [12, 13] that minimizes $|O_1 - O_2 R|$. If this works, then we can support incremental updates by rotating M_2 to be compatible with M_1. Of course, a single global rotation will not fit the data all that well. It may be preferable to model the problem as a piece-wise linear interpolation with a number of local rotations. If this proves unsatisfactory, it may be necessary to consider more sophisticated methods in machine learning.

Four Use Cases

1. Recommendations: What should I read? What should I cite?
2. Finding experts [14, 15, 16]: Who knows [17]? Finding experts is similar to recommendations for papers, except the answer should be a contact, as opposed to a paper.
3. Routing submissions to reviewers [18, 19, 20]: conferences, journals and funding agencies use reviewing platforms [21, 22, 23, 24]. Many of these platforms use software to suggest who should review what based on abstracts and titles. These systems should take advantage of references. People cited in a submission are likely to be qualified to review (and interested in the topic). There are opportunities to improve assignments, as well as evaluation benchmarks.
4. Summarization of sets of documents, emphasizing pairwise similarities between documents (unlike tl;dr methods, which process each document one at a time, missing pairwise relations).

Similarity plays a key role in these use cases. There is an opportunity for creative ensembles of deep nets and spectral clustering [25, 26, 27]. Deep nets have been effective for capturing similarities of texts; spectral clustering is promising for links. Tables 1-2 compare two estimates of similarity, cos_S and cos_P. The former is based on Specter [5], a deep net based on SciBERT [28], and fine-tuned with citations. The latter is a spectral clustering of the citation graph, G. Documents with large cos_S have similar abstracts; documents with large cos_P are near one another in G.

Use Case 1: Recommendation Systems

It is becoming increasingly difficult to keep up with the literature. Suppose you found one paper in a "hot" area. Can we use that paper as a query, q, to find more? Tables 1-2 start with two qs:

- Table 1, q =[29]: "DeepWalk: online learning of social representations" [25]
- Table 2, q = [30]: "On the Dangers of Stochastic Parrots..." [31]

f	$cos_S(q,c)$	$cos_P(q,c)$	**Cites**	**Paper**
f_S	0.984	0.827	0	Learning of Social Representations [32]
f_S	0.809	0.951	2	Topic-aware latent models for representation learning on networks [33]
f_S	0.797	0.947	4	SimWalk: Learning network latent representations with social relation similarity [34]
f_S	0.783	0.033	0	Deep Representation Learning on Complex Graphs [35]
f_P	0.771	0.999	6007	node2vec: Scalable Feature Learning for Networks [36]
f_P	0.711	0.998	3632	LINE: Large-scale Information Network Embedding [37]
f_P	0.664	0.997	1025	A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications [38]
f_P	0.711	0.996	1157	metapath2vec: Scalable Representation Learning for Heterogeneous Networks [39]

Table 1: Approximate n-best matches for query, q = [29]: "DeepWalk..." [25]. Four candidates, c, are shown for f_S (Specter), followed by four for f_P (Proposed). The latter have more citations.

The two tables compare cos_S and cos_P. Both methods use approximate nearest neighbors (ANN) [46, 47, 48] to find recommendations with large cosine scores. The recommendations above the line have large cos_S, whereas the recommendations below the line have large cos_P. The tables show

f	**Rank**	$cos_S(q,c)$	$cos_P(q,c)$	**Citations**	**Paper**
f_S	1	0.794	0.957	3	What Language Model to Train if You Have One Million GPU Hours? [40]
f_S	2	0.779	0.976	117	On the Stability of Fine-tuning BERT: Misconceptions, Explanations, and ... [41]
f_S	3	0.777	0.961	1	Emergent Properties of Finetuned Language Representation Models [42]
f_P	1	0.615	0.992	139	RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models [43]
f_P	2	0.750	0.992	49	The Radicalization Risks of GPT-3 and Advanced Neural Language Models [44]
f_P	3	0.542	0.992	52	Persistent Anti-Muslim Bias in Large Language Models [45]

Table 2: Approximate n-best matches for query, q = [30]: "On the Dangers of Stochastic Parrots..." [31], a paper on Responsible AI aspects of deep nets. All six candidates are about deep nets, but f_P candidates have more citations, and they are more about toxicity and Responsible AI.

both cosine scores for all recommendations so one can see appreciate the similarities and differences. This redundancy will be used to deal with realities such as missing values and bad data.

There is a considerable literature on definitions of relevance [49, 50, 51]. Many recommendation systems focus on "relevance" (overlap in words between queries and candidates), but in addition to that, we should also consider credibility. The recommendations in Tables 1-2 are all on deep nets, but the proposed method recommends papers with more impact (citations). In general, we should recommend papers based on behavioral signals (citations, page views and downloads). This work will focus on citations because that behavioral signal is more available and less sensitive. As we will see, there will also be applications of these methods in NLP tasks such as summarization.

More Use Cases

Suppose you are writing a paper and you just made an assertion that should be backed up with a citation. Could you highlight the assertion and ask the system to suggest some candidate citations? Is this process invertible? Suppose we start with a set of papers. Can the system suggest some assertions that calls out similarities and differences among the papers?

Suppose you are writing a paper and you suspect you may be missing some important references. Could you highlight a few references and ask the system to suggest some more? It is likely that the system will find too many relevant papers. How do you know where to begin? We could use a method for summarizing a large set of papers. How are this year's ACL papers differ from last year's? Similarly, compare and contrast papers by the Bengio brothers. How do Samy's papers differ from his brother's? Many systems for summarizing documents process each paper, one by one. Suppose we want to identify clusters of papers, and compare and contrast similarities and differences among clusters.

Finding references is like finding experts. In large enterprises, it is hard to know who knows what. A number of systems have been built where the answer is a contact instead of a document [14, 15, 16, 17]. Routing submissions to reviewers is related to finding experts.

Would it be possible for a system to write much of the "related work" section of a paper automatically [52]? Ideally, after reading a well-written "related work" section, the reader should

know where the paper is going, and what the contribution will be. It may be asking too much of the system to anticipate the punch line, but we should be able to build a system that will return a comprehensive set of related papers. The system could also cluster the papers into topics, and suggest some ways to organize the literature into larger structures such as timelines, topics, schools of thought, and directions forward.

Systems of this kind could also help reviewers. Reviewers are often asked about missing references. We could even imagine our project creating a set of tools to be used by both authors and reviewers to improve their work, and to prepare for feedback from the other.

One could think of queries as prompts. It has become popular recently to engineer prompts for ChatGPT [53]. So too, one could engineer queries for recommendation systems.

Use Cases 2-3: Finding Experts And Reviewers

Conferences and funding agencies use platforms such as softconf, easychair and openreview [21, 22, 23, 24]. Some platforms use software to assign papers to reviewers, though we need better assignments (and better evaluations). We fear that bad assignments are teaching authors to write incremental papers that can be reviewed by unqualified and unsympathetic reviewers [54]. Better assignments will improve the quality of the scientific literature. The citation graph can help find reviewers that are familiar with the background references, and sympathetic with the area.

$$\hat{f}(i) \approx \sum_{j \in (i,\cdot)} f(j) \qquad (1)$$

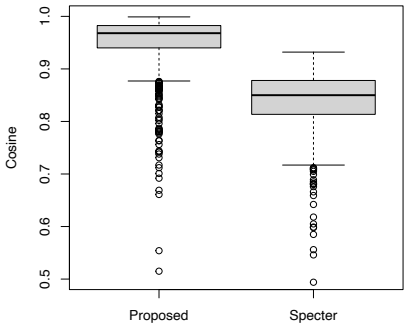


Figure 3: $f(i) \approx \hat{f}(i)$

Use cases 2-3 (finding experts and reviewers) are similar to case 1 (recommendations), except that candidates should be people (experts/reviewers) as opposed to papers. There are some interesting challenges for mapping between people and vectors. Perhaps the simplest starting point is to represent authors by the centroids of their papers.

In some use cases (use case 1 and 2 for recommending papers and finding experts), we can assume the queries are members of the documents we have downloaded from [4], but this assumption is not appropriate for assigning submissions to reviewers (case 3), because submissions are unlikely to be in the document collection of (mostly) published papers. For finding reviewers, let us assume the query is a submission, which contains a number of potentially useful fields such as: a title, abstract, authors, body and references. Previous work focuses on titles and authors, though other fields should be considered. Some fields may be inappropriate, such as authors (in double-blind contexts). Full-text may be challenging for BERT-based models with a limited window of 512-subword units. We expect references to be easier (and more rewarding).

Applying Specter to submissions is relatively straightforward; we can follow the suggestion in [55] and simply apply the Specter model on HuggingFace to titles and abstracts. But how do we use the proposed method to map submissions to vectors? Thus far, we have applied the proposed method to papers in the citation graph. How do we estimate vectors for submissions, assuming submissions are not in the graph? Eqn (1) suggests a simple way forward. That is, the vector for paper i is close the the centroid of i's references. We hope to improve over the centroid approximation, but it is a reasonable place to start.

Figure 3 tests the centroid approximation on a set of 983 papers. For each of these, we have $f(i)$, and we obtain, $\hat{f}(i)$ by applying Eqn (1) to the references of i. The boxplots in Figure 3

show $cos(f(i), \hat{f}(i))$. The cosines are large, especially for the proposed method. These large cosines suggest the centroid approximation is often effective, especially for the proposed method. GNNs use links for fine-tuning, but there may also be an opportunity to use links at inference time.

Coping With Realities: Missing Values And Bad Data

Multiple representations introduce possibilities for dealing with realities such as missing values, as illustrated in Figure 4. Some of the 208M papers in [4] have abstracts (A) and some have links (L). Specter is applicable for $|A| \approx 16\% + 31\% \approx 47\%$, and the proposed method is applicable for $|L| \approx 22\% + 31\% \approx 53\%$. An ensemble of both methods increases coverage to $|A \cup L| \approx 70\%$.

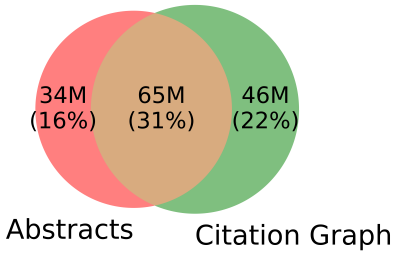


Figure 4: Venn diagram of [4]

Multiple representations also create possibilities for detecting bad data. The data from Semantic Scholar [4] is extremely useful, but it is not free from errors. Bad data is often associated with large differences between cos_P and cos_S. In Table 1, cos_P is too low for candidate [35] because of gaps in the citation graph. Table 3 shows some examples where cos_S is too high because of bad data in the data structure for abstracts. Most of the abstracts in Semantic Scholar are correct, but there are a few documents in Semantic Scholar where the abstract field in the database was incorrectly replaced with some boilerplate from JSTOR:

> *JSTOR is a not-for-profit service that helps scholars, researchers, and students discover, use, and build upon a wide range of content in a trusted digital archive...*

When this error happens to the query, then the recommender system illustrated in Tables 1-2 is likely to find a number of candidates with similar errors, as shown in Table 3. In this case, the error impacted the query [56] as well as the candidates [57, 58, 59], producing large (misleading) cos_S because the"abstracts" (according to the database) are nearly identical, though obviously, these papers have little in common. Fortunately, these errors can be detected by comparing cos_S with cos_P, because different errors impact different cosines differently; cos_S is sensitive to errors in abstracts whereas cos_P is sensitive to errors in the citation graph. Multiple redundant estimates of similarity create opportunities for robustness.

f	**Rank**	cos_S	cos_P	**Citations**	**Paper**
f_S	7	0.777	0.097	15	The Value of Natural Sounds [57]
f_S	8	0.766	0.089	16	Re-Examining the Foundations [58]
f_S	9	0.757	0.106	8	Facts and Faith in Biblical History [59]

Table 3: Multiple embeddings create opportunities for error detection. Normally, $cos_S \approx cos_P$, but errors in the abstract database can produce inflated estimates of cos_S. Errors can often be detected by comparing cos_S and cos_P, because different errors impact different cosines differently.

Notation

1. Let $N \approx 200$M be the number of documents in the collection.
2. Let $E \approx 2$B be the number of citations in the collection.
3. Let K be the number of hidden dimensions (typically, $K = 768$ for models based on BERT, and $K = 280$ for models based on Spectral Clustering and Linear Algebra).

4. Let f be a model (such as BERT) and M be an embedding. Models f can be applied at inference time to novel inputs, or they can be rows of M. Embeddings, M, are large matrices: $M \in \mathbb{R}^{N \times K}$, where $f(d_i)$ is the row in M for document d_i. That is, $f(d_i) = M[d_i,]$.
5. Let f^{-1} be the inverse of f. f maps document ids to vectors ($f(d_i) = M[d_i,]$), and f^{-1} maps vectors to document ids ($f^{-1}(M[d_i,]) = d_i$). We can implement f^{-1} with approximate nearest neighbors (ANN) [46, 47, 48], so f^{-1} can be applied to vectors that do not match any of the rows in M exactly. (We refer to f^{-1} as "BERT-inverse.")
6. The subscripts S and P will be used as necessary to distinguish f_S from f_P, and M_S from M_P. The subscript S refers to Specter, and the subscript P refers to the model proposed here, which is based on ProNE [27]. More subscripts will be introduced for additional embeddings.

More Embeddings

We will experiment with many embeddings, in addition to M_S and M_P. One such embedding is based on citing sentences [60, 61, 62], which are like anchor text [63] in web search. We expect citing sentences to be helpful when subsequent literature introduces contemporary terminology. Consider, for example, Turing's seminal paper [11], which introduced what is now known as a "Turing Machine," though of course, Turing did not use that term in his paper. The term is very common in sentences that cite his paper. About one-third of them mention Turing Machines. More generally, it is common for the subsequent literature to name important concepts after important figures in the literature, but it is rare to find the modern terminology in the primary source. We have already run Specter on a billion citing sentences, and hope to show that citing sentences are helpful. Citing sentences might be even more useful than primary sources for appreciating contributions in contemporary contexts.

In addition to citing sentences, there are a number of alternatives to Specter in the literature including Specter2 [64], Link BERT [65] and SciNCL [66]. The Semantic Scholar API provides convenient access to Specter embeddings [67], but their API does not currently support alternatives. We will run these alternative models on as many papers as possible and distribute the results on Globus [68]. We will also distribute indexes for approximate nearest neighbor (ANN) search [46, 47, 48].

Note that these files are large. Embeddings are dense matrices, $M \in \mathbb{R}^{N \times K}$, where $N \approx 200$M and $K \approx 768$. If each value in M is a 4-byte float, then each embedding is 614GBs. The indices are relatively small, though probably too large for GitHub. Each index is a permutation of N (about a GB). We often use a dozen or more indexes per embedding.

Indexes For Approximate Nearest Neighbors (ANN): Permutations On N

The indexes implement ANNs in external memory. That is, each index is a permutation, π on N, constructed so vectors near one another in π have large cosines. Thus, $cos(M[\pi(i), \cdot], M[1+\pi(i), \cdot])$ is large. The computation of π is embarrassingly parallel, and requires remarkably little RAM. The inputs are the embedding, $M \in \mathbb{R}^{N \times K}$, a random seed, and $b \approx 48$, the number of random bits to compute. The process converts the embedding, M to bit vectors $B \in \{0,1\}^{B \times K}$ in the usual way, where cosines in M are related to Hamming distances in B. However, we do not need to load M into memory, or keep the bit vectors for long since they can be recomputed later from the seed, but we often do not need them. The random vectors $V \in \mathbb{R}^{b \times K}$ are computed from the seed. We compute B by streaming vectors, m, from M. The j^{th} bit for m is $m \cdot V[j, \cdot] > 0$. The output index, π, is the argsort of B. After saving π, we can discard V and B, since they can be recomputed from the seed, if necessary. Since this process is embarrassingly parallel, multiple indexes can be

computed in separate batch jobs.

At query time, we memory map M and the indexes, so the process starts up quickly without having to load large files into memory. The ANN process uses the indexes to find candidates near the query. Each candidate is scored by computing cosines of the appropriate rows in M. Code for creating indexes and using them at inference time has been posted on GitHub [69]. The GitHub provides pointers to large files on Globus [68] for M_S and M_P with indexes.

Cos-Dist: Multiple Views Of Similarity

Cosines in embeddings can be used to estimate distances in graphs, and vice versa. We view embeddings, M, and graphs, G, as different representations of documents, somewhat analogous to frequency and time in speech. Operations such as filtering can be implemented as multiplication in the frequency domain, or convolution in the time domain. So too, under Cos-Dist, one can estimate document similarities with cosines in the embedding space and distances in graphs. Different representations have different advantages and disadvantages. Embeddings are convenient for estimating cosines, and finding approximate nearest neighbors, though graphs are more compact.

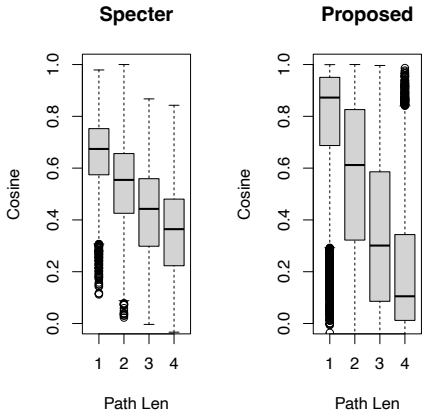


Figure 5: *Cos-Dist*: Cosines in M can be interpreted as distances in G.

Graphs, $G = (N, E)$, are more compact than embeddings, $M \in \mathbb{R}^{N \times K}$, because G is stored as a sparse matrix and M is dense. Storage for G depends on E, where $E \ll NK$. Recall $E \approx$ 2B citations, $N \approx$ 200M papers, and $K = 280$ for proposed method ($K = 768$ for Specter).

Prior work encodes documents as vectors in just one way, f. We propose multiple perspectives, f_1, f_2,..., to capture titles, abstracts, full text, citations and other properties of documents. Let $cos(f(i), f(j))$ denote the similarity of i and j. Cosines of vectors based on text (e.g., bags of words [70, 71, 72], BERT [1, 5, 28]) denote word similarity, whereas cosines based on spectral clustering of citation graph, G [73, 27] can be interpreted in terms of $dist(i, j)$, distances in G.

We observe that documents that are similar in at least one way, tend to be similar in other ways, as well. We refer to this assumption as Cos-Dist: $cos(f_1(i), f_1(j)) \sim cos(f_2(i), f_2(j)) \sim dist(i, j)$. Diversity over representations opens many opportunities:

1. Interpretability (Cos-Dist): Similarity of documents can be estimated as cosines in embeddings, $cos(f(i), f(j))$, or distances in G. In other words, embeddings can be viewed as an alternative representation of graphs. Both representations have advantages and disadvantages. Graphs are compact, but embeddings are convenient for computing cosines and ANNs.
2. Redundancy: if we have two embeddings, f_1 and f_2, then we have three redundant estimates of document similarity: (a) $cos(f_1(i), f_1(j))$, (b) $cos(f_2(i), f_2(j))$ and (c) distances in G. Redundant estimates create opportunities for error detection and error correction.
3. Cos-Dist holds for embeddings based on Linear Algebra as well as embeddings based on non-linear deep nets, creating opportunities to generalize results from relatively well understood Linear Algebra to recent advances in deep nets, as suggested in [7].
4. Diversity over representations introduces opportunities for diverse computing environments. For example it has been popular to use GPUs with GBs of RAM for deep nets, but CPUs

with TBs of RAM may be preferable for embeddings based on Linear Algebra. When we have TBs of RAM, it becomes feasible to compute the SVD of a large citation graph in RAM, as will be done in the prefactorization step of ProNE. Standard recipes for training deep nets with mini-batches can be viewed as an external memory algorithm. Because external memory is slower than RAM, CPUs with TBs may be preferable to GPUs with GBs, at least for some memory-bound computations.

Figure 5 provides some evidence for the Cos-Dist assumption (distances in citation graph are related to cosines in embeddings). We created pairs of documents with a random walk. We started with nearly 1M document ids, selected at random. From each of these, we randomly selected one of its references. From there, we continued the walk by randomly selecting one of its references, and so on. For each of these pairs, when the cosine is available, we report the length of the shortest path on the x-axis, and cos_S (and cos_P) on the y-axis. More than 20% of the pairs are omitted from the plot for Specter because those cosines are often unavailable (due to missing abstracts).

For both Specter and the proposed method, we find that cosines are negatively correlated with path lengths, though the pattern is stronger for f_P because that method uses an optimization criterion that is closer to Cos-Dist. We believe, however, that Cos-Dist holds for most reasonable embeddings of documents because they are all estimating similarities of the same documents.

ProNE

The ProNE method, f_P, is based on the citation graph, G. We start by downloading G from [4]. G is stored as an adjacency matrix, an $N \times N$ Boolean matrix, $M_G \in \{0,1\}^{N \times N}$, with $N \approx 200$M nodes (papers). If (and only if) paper d_i cites paper d_j, then $M_G[d_i, d_j] = 1$. This matrix is sparse because there are only $E = 2$B citations (nonzero edges). G might appear to be a huge matrix but it requires just 7.8 GBs when stored in scipy [74] as an npz file because G is extremely sparse.

We then run the ProNE [27] method in nodevectors [75] to produce an embedding, $M_P \in \mathbb{R}^{N \times K}$, where N is the number of nodes (papers) in G, and K is the number of hidden dimensions. ProNE is a variant of node2vec [73]. Unlike Specter and many variants of node2vec, ProNE is based on Linear Algebra and spectral clustering, and does not use deep nets.

A work item (below) is to make it easier to compute M_P. ProNE takes nearly a week to compute M_P (for $K = 280$) on a CPU with 2 TBs of RAM. The first step is called *prefactorization*. It uses a memory internal SVD procedure to factor a normalized version of G as UDV^T, where $U, V \in \mathbb{R}^{N \times K}$. It is helpful to have TBs of RAM to compute the SVD because U and V are large dense matrices (unlike G, which is a sparse matrix). After prefactorization, ProNE uses spectral propagation to compute the embedding, M_P, so cosines in M_P can be interpreted in terms of distances on G. For efficiency, ProNE uses Chebyshev expansion to avoid explicit Eigen decomposition. The Chebyshev expansion consumes even more memory than SVD, because it uses several dense matrices with the same size as U. ProNE does not currently support incremental updates, though adding that feature would make it easier to keep M_P up to date.

Use Case 4: Document Summarization

Tables 1-2 suggest another NLP application: summarization [52, 76, 77, 78, 79]. In addition to titles (as in Tables 1-2), it would be desirable to produce snippets/summaries to help users choose among the various document recommendations. It is common for summarization systems to process each document one at a time, though it would be desirable for summaries to compare and contrast the choices to help users appreciate similarities and differences. A work item (below) is to produce summaries that improve over two baselines: (1) tl;dr from Semantic Scholar [4], and (2) summaries

Title	tl;dr (too long; didn't read) Summaries
On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? [30]	Recommendations including weighing the environmental and financial costs first, investing resources into curating and carefully documenting datasets rather than ingesting everything on the web, and carrying out pre-development exercises evaluating how the planned approach fits into research and development goals and supports stakeholder values are provided.
RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models [43]	It is found that pretrained LMs can degenerate into toxic text even from seemingly innocuous prompts, and empirically assess several controllable generation methods find that while data- or compute-intensive methods are more effective at steering away from toxicity than simpler solutions, no current method is failsafe against neural toxic degeneration.
The Radicalization Risks of GPT-3 and Advanced Neural Language Models [44]	GPT-3 demonstrates significant improvement over its predecessor, GPT-2, in generating extremist texts and its strength in generating text that accurately emulates interactive, informational, and influential content that could be utilized for radicalizing individuals into violent far-right extremist ideologies and behaviors.
Persistent Anti-Muslim Bias in Large Language Models [45]	GPT-3, a state-of-the-art contextual language model, captures persistent Muslim-violence bias, demonstrating that it appears consistently and creatively in different uses of the model and that it is severe even compared to biases about other religious groups.

Table 4: tl;dr summaries from Semantic Scholar [4] for some papers from Table 2.

from ChatGPT [53]. Table 4 illustrates baseline (1) for three candidates from Table 2. These summaries tend to be extracts from abstracts. To illustrate baseline (2), we ran ChatGPT on the text in Table 4 and it produced the following output:

> *The recent research on the dangers of large language models (LLMs) highlights the importance of considering their potential negative impacts on society. As language models become increasingly sophisticated, there is a risk of them degenerating into toxic text or perpetuating biases and harmful ideologies. Therefore, it is crucial to evaluate the risks and benefits of LLMs before developing and deploying them.*
>
> *One recommendation for addressing the risks of LLMs is to weigh the environmental and financial costs of building and training these models. LLMs are known to require large amounts of energy and computational resources, which can have significant environmental impacts...*

ChatGPT has a tendency to hallucinate. Neither baseline captures relations (comparisons/contrasts) across documents. We suggest Eqn (2), where d_a and d_b are documents with vectors: $a = f(d_a)$ and $b = f(d_b)$, respectively. C is a set of candidate vectors such vectors in M near a.

$$\begin{aligned} compare(a,b) &= ARGMIN_{c \in C}\ |c-a| + |c-b| \\ contrast(a,b) &= ARGMIN_{c \in C}\ |c-a| - |c-b| \end{aligned} \tag{2}$$

We can then use $f^{-1}(compare(a,b))$ and $f^{-1}(contrast(a,b))$ to find text and documents from vectors of interest. We think of f^{-1} as "BERT-inverse." From documents, we can report properties of interest such as titles, abstracts, tl;dr, references, citations, and more.

Data, Applications & Evaluations

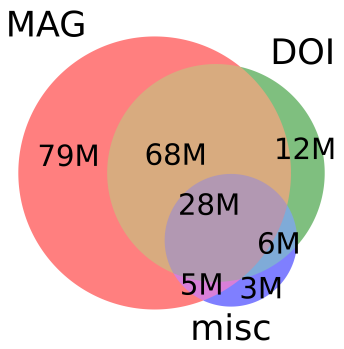


Figure 6: MAG [80], DOI [81] plus misc (PubMed, PubMedCentral, DBLP, arXiv, ACL)

For evaluations, we plan to start with benchmarks on GitHub [82, 83]. Baseline results are reported in [5, 84]. These benchmarks are based on 200M papers in [4].

Figure 6 shows the importance of MAG (Microsoft Academic Graph). Semantic Scholar is based on two large sources (MAG [85, 80], DOI [81]) plus five smaller sources labeled *misc* (PubMed [86], PubMedCentral [87], DBLP [88], arXiv [89] and ACL [90]). Misc contributes 3M papers beyond MAG ∪ DOI. Much of the work in our field is on the ACL Anthology and arXiv, a relatively small subset of misc.

The 200M papers cover a diverse set of fields of study (fos), though with an emphasis on STEM: Medicine (45M), Chemistry (13M), Computer Science (13M), Biology (13M), Materials Science (10M), Engineering (8M), Physics (7M), Psychology (7M), Mathematics (5M), Political Science (4M), Business (4M), Sociology (3M), Geography (3M), Economics (3M), Environmental Science (3M), Geology (3M), History (2M), Art (2M), Philosophy (1M). We have found other sources such as SSRN (Social Science Research Network) [91] to be more useful for Social Studies, Law and Journalism, even though SSRN is relatively small (1M papers). SSRN also makes use of behavioral signals on downloads, which is extremely useful, especially in fields where it is less common for papers to cite one another. Unfortunately, SSRN does not make its data available for bulk download.

Another benchmark is OGB (Open Graph Benchmark) [92, 93] from 2021 KDD-Cup [6]. The literature on graph learning classifies tasks into (a) node-level, (b) link-level and (c) graph-level. The example of (a) in [93] is closely related to tasks of interest for this proposal. We will evaluate our work on task (a) in OGB, but we need to be careful when reporting results, since we intend to use outside data [4], a violation of OGB rules. In addition, we are interested in practical applications. Since OGB was designed for another purpose, they chose to keep it a secret how ids map to strings of interest (paper titles, author names, subject areas). This choice discourages cheating (use of outside data), but it also makes it necessary to retrain methods developed for OGB on outside data before such methods can be deployed for use in practical applications.

OGB uses MAG240M for (a). MAG240M is based on a snapshot of MAG from a few years ago before MAG was retired and transferred to the nonprofit OpenAlex [94]. Semantic Scholar ($N \approx$ 200M papers and $E \approx$ 2B citations) is now almost twice as large as MAG240M ($N \approx$ 122M papers and $E \approx$ 1.3B citations). MAG240M has labels for a relatively small set of just 1.4M arXiv papers. The task is to predict these labels (about 150 subject areas from arXiV).

There are a few similar tasks in [83] such as: fos (field of study), and MeSH descriptors. These tasks are more attractive because they provide a mapping from numeric label identifiers to meaningful strings. Moreover, there are only 2M arXiv papers, and it is not clear that subject area labels for these papers will generalize well to papers from other sources. There are many more papers in Semantic Scholar with fos labels (148M), and these labels are easier to interpret. The MeSH task is attractive since the National Library of Medicine has been labeling PubMed papers with MeSH terms for many years. PubMed (35M papers) is also considerably larger than arXiv.

As for link prediction, there are a couple of tasks in [83] such as cite prediction that are attractive because they generalize to important practical problems. Recommender systems should be able to answer questions such as what should I read and what should I cite [95, 96, 97, 98]. The cite prediction task, as described in [5], provides pairs of document identifiers with titles and abstracts.

The pairs are labeled with 1 or 0, indicating whether the first paper of the pair cites the second member of the pair or not. The task is to predict the label. Unfortunately, the description in [5] does not match the benchmark [83]. We will report results on the benchmark, as well as a corrected version with labels that match the description in [5],

The proposed work will improve over evaluations above in two respects. First, most of the evaluations above make use of titles and abstracts and little else. We plan to take advantage of additional features (citations, citing sentences and full text) in a way that is robust to missing/corrupted values. In addition, the evaluation should guide the community toward building effective solutions to practical tasks such as the four use cases mentioned above: (1) recommendations, (2) finding experts, (3) routing submissions to reviewers and (4) summarization. We will propose evaluations that are relevant to many of these use cases:

1. Label Prediction: predict labels such as subject areas, fos (field of study), MeSH.
2. Link Prediction for Recommender systems: what should I read? what should I cite?
3. Search / Rank Retrieval / Relevance Feedback: given a query (text and/or documents with relevance labels): find documents in the collection that are similar to the query.
4. Paper Reviewer Matching: given a paper (text and/or document id), identify the best set of reviewers for the paper, satisfying a variety of constraints including topic expertise, avoiding conflicts of interest, diversity of reviewer institutions, and load balancing.
5. Summarize collection of documents: compare and contrast clusters (as opposed to summarizing each document in isolation).

Results From Prior NSF Support

Neither PI Church nor Co-PI Chandrasekar have received NSF funding in the past five years, since they have been in industry. Church was a Co-PI for NSF #1040114 when he was at JHU in 2010.

Deliverables

1. Create and disseminate tools that make it easier for researchers to make better use of the scientific literature.
2. Resources: embeddings, models and code. We will share large files such as embeddings of 10^8 papers on Globus [68]. Code will be posted on GitHub. Models and benchmarks will be posted on HuggingFace. Some code and embeddings have been posted already on [69].
3. Timeliness: Support incremental updates to resources.
4. Use cases: (1) recommendations, (2) finding experts, (3) routing submissions to reviewers and (4) summarization.
5. Evaluation: Better benchmarks, as well as better numbers on established benchmarks.
6. Establish that it is important to experiment with large (and growing) graphs to appreciate network effects (Metcalfe's Law). Also, establish the value of multiple perspectives in theory and practice. We will generalize results from linear algebra to deep nets, as mentioned above. From a practical perspective, combinations of text and links create opportunities to deal with missing values and bad data.

Schedule/GANTT Chart

Figure 7 presents a GANTT chart for the work items in the proposal.

	Beyond Titles and Abstracts: Text + Links are Better Together				PROJECT TIMELINE (in months, starting Jan 2024)																				
					Year 1												Year 2								
#	ACTIVITY	PLAN START (month #)	Start if project begins Jan 2024	PLAN DURATION (months)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	Create new embeddings of documents	1	Jan-24	9																								
2	Develop methods to recommend literature	5	May-24	6																								
3	Evaluate all embeddings	7	Jul-24	4																								
4	Make computing ProNE embeddings easier	9	Sep-24	4																								
5	Develop techniques to find experts, starting from vector centroid approximation	10	Oct-24	4																								
6	Develop better ways to assign papers to reviewers	12	Dec-24	6																								
7	Work on theoretical framework to unify deep nets and linear algebra	14	Feb-25	7																								
8	Develop methods to compare, contrast and summarize clusters of documents, with similarities & differences	17	May-25	6																								
9	Prepare final report	22	Oct-25	3																								
10	Mentor students and post-doc(s)	1	Jan-24	24																								

Figure 7: GANTT Chart for proposal

Accomplishments Thus Far, Risks And Contingencies

PI Church is leading a team on this topic at the 6-week 2023 Jelinek Summer Workshop on Speech and Language Technology [99]; some code and resources have already been posted on [69]. The

deliverables in the previous section go well beyond what has already been done, and what will be accomplished during the Jelinek Summer Workshop. The long list of deliverables is a stretch goal. There is a risk that it may become necessary to make adjustments.

Broader Impacts

There are many opportunities for applications of the deliverables in academia and industry, including web search, product recommendations and traffic analysis (data mining on telephone call detail and internet packet headers). The proposed work will make it it easier for the community to make better use of the scientific literature.

Improving access to expertise will produce significant benefits to people in resource-poor countries and environments. It will be easier for researchers in diverse settings to contribute to the literature in meaningful ways. Science will advance in more productive ways when the right people can more easily see how they can contribute to critical projects.

There are opportunities to trial the tools that we will develop with students at Northeastern University. Northeastern has large numbers of excellent students. Many are the first in their family to attend college. Many are members of protected classes. Of course, before running experiments with human subjects, we will obtain required permission from Northeastern's IRB.

Northeastern has many campuses in many locations including Mills College in Oakland, California and the Roux Institute in Portland, Maine. Both locations offer opportunities to make the scientific literature more accessible to more diverse communities.

It would be interesting to see how effective the proposed tools are with users from more diverse backgrounds. We have already begun experimenting with these tools with users from a variety of research fields. So far, we have discovered opportunities for improving coverage in fields that go beyond STEM such as journalism and the law.

PI Qualifications

Church and Chandrasekar have considerable experience in computational linguistics and related fields, as indicated by their h-indexes on Google Scholar [100, 101]. They both worked at Microsoft on Bing, and have considerable experience with behavioral signals such as web search logs [102, 103]. They have also worked with large databases of telephone call detail at AT&T Bell Labs and elsewhere [104]. Call detail is often used for traffic analysis in intelligence applications where much can be inferred from the graph of who is communicating with whom. Data mining on call detail is similar to what we propose to do with citation graphs.

Church was an early advocate in 1990s of the revival of empirical methods and corpus-based lexicography. His most cited paper introduced what is now known as PMI (point-wise mutual information) [105]. Levy and Goldberg [106] established a connection between PMI and Word2Vec. There are clear connections between [106] and more recent methods such as deep nets (BERT) [1]. Qiu et al. [7] use methods like [106] to unify GNNs and spectral clustering. Church was also a founder of EMNLP and served as president from 1993 to 2011 of the ACL special interest group (SIGDAT) that runs EMNLP. Church was president of the ACL in 2012. He was an AT&T Fellow in 2001, ACL Fellow in 2015 and Baidu Fellow in 2018. Baidu is the largest web search company in China.

References Cited

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. [Online]. Available: https://aclanthology.org/N19-1423

[2] anonymous, "Graph neural networks," https://snap-stanford.github.io/cs224w-notes/machine-learning-with-networks/graph-neural-networks, 2020.

[3] "Google scholar," https://scholar.google.com/.

[4] Semantic-Scholar, "Semantic scholar academic graph api: Providing a reliable source of scholarly data for developers," https://www.semanticscholar.org/product/api, 2017.

[5] A. Cohan, S. Feldman, I. Beltagy, D. Downey, and D. Weld, "SPECTER: Document-level representation learning using citation-informed transformers," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 2270–2282. [Online]. Available: https://aclanthology.org/2020.acl-main.207

[6] W. Hu, M. Fey, H. Ren, M. Nakata, Y. Dong, and J. Leskovec, "Kdd cup OGB large-scale challenge (OGB-LSC) session 1," https://www.youtube.com/watch?v=MblxCqwengI, 2021.

[7] J. Qiu, Y. Dong, H. Ma, J. Li, K. Wang, and J. Tang, "Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec," *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 2017.

[8] R. Van Noorden, "Global scientific output doubles every nine years," *Nature news blog*, 2014.

[9] P. Fisk, "Metcalfe's law explains how the value of networks grows exponentially ... exploring the "network effects" of businesses like apple, facebook, trulia and uber," https://www.peterfisk.com/2020/02/metcalfes-law-explains-how-the-value-of-networks-grow-exponentially-there-are-5-types-of-network-effects/.

[10] B. Metcalfe, "Metcalfe's law after 40 years of Ethernet," *Computer*, vol. 46, no. 12, pp. 26–31, 2013.

[11] A. M. Turing, "On computable numbers, with an application to the entscheidungsproblem," *J. of Math*, vol. 58, no. 345-363, p. 5, 1936.

[12] J. C. Gower and G. B. Dijksterhuis, *Procrustes problems*. OUP Oxford, 2004, vol. 30.

[13] "scipy.linalg.orthogonal_procrustes," https://docs.scipy.org/doc/scipy/reference/generated/scipy.linalg.orthogonal_procrustes.html.

[14] D. Yimam-Seid and A. Kobsa, "Expert-finding systems for organizations: Problem and domain analysis and the demoir approach," *Journal of Organizational Computing and Electronic Commerce*, vol. 13, no. 1, pp. 1–24, 2003.

[15] M. T. Maybury, "Expert finding systems," https://www.mitre.org/sites/default/files/pdf/06_1115.pdf, 2006.

[16] Y. Tu, N. Johri, D. Roth, and J. Hockenmaier, "Citation author topic model in expert search," in *Coling 2010: Posters*. Beijing, China: Coling 2010 Organizing Committee, Aug. 2010, pp. 1265–1273. [Online]. Available: https://aclanthology.org/C10-2145

[17] L. A. Streeter and K. E. Lochbaum, "Who knows: A system based on automatic representation of semantic structure," in *User-Oriented Content-Based Text and Image Handling*, 1988, pp. 380–388.

[18] D. Mimno and A. McCallum, "Expertise modeling for matching papers with reviewers," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2007, pp. 500–509.

[19] https://mimno.infosci.cornell.edu/data/nips_reviewer_data.tar.gz, 2007.

[20] R. Kinney, "Conference Peer Review with the Semantic Scholar API," https://blog.allenai.org/conference-peer-review-with-the-semantic-scholar-api-24ab9fce2324, 2021.

[21] https://openreview.net/.

[22] https://www.softconf.com/.

[23] https://easychair.org/.

[24] https://cmt3.research.microsoft.com/.

[25] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 701–710.

[26] S. Cao, W. Lu, and Q. Xu, "GraRep: Learning graph representations with global structural information," in *Proceedings of the 24th ACM international on conference on information and knowledge management*, 2015, pp. 891–900.

[27] J. Zhang, Y. Dong, Y. Wang, J. Tang, and M. Ding, "ProNE: Fast and scalable network representation learning." in *IJCAI*, vol. 19, 2019, pp. 4278–4284.

[28] I. Beltagy, K. Lo, and A. Cohan, "SciBERT: A pretrained language model for scientific text," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 3615–3620. [Online]. Available: https://aclanthology.org/D19-1371

[29] "Deepwalk: online learning of social representations," https://www.semanticscholar.org/paper/fff114cbba4f3ba900f33da574283e3de7f26c83.

[30] E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell, "On the dangers of stochastic parrots: Can language models be too big?" https://www.semanticscholar.org/paper/On-the-Dangers-of-Stochastic/6d9727f1f058614cada3fe296eeebd8ec4fc512a.

[31] ——, "On the dangers of stochastic parrots: Can language models be too big?" in *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 2021, pp. 610–623.

[32] "Learning of social representations," https://www.semanticscholar.org/paper/93b050f5bf0567a675979cd564cbe66ff9c3a78f.

[33] "Topic-aware latent models for representation learning on networks," https://www.semanticscholar.org/paper/21ee2cc0bf41c1b74efb6104edd4df73416b46c1.

[34] "Simwalk: Learning network latent representations with social relation similarity," https://www.semanticscholar.org/paper/e294339b402ce055d5a5198becc35b2dbbd20a9a.

[35] "Deep representation learning on complex graphs," https://www.semanticscholar.org/paper/bb11bec51c2e069ef0ddba4eb3117c9dbc8a4584.

[36] "node2vec: Scalable feature learning for networks," https://www.semanticscholar.org/paper/36ee2c8bd605afd48035d15fdc6b8c8842363376.

[37] "LINE: Large-scale information network embedding," https://www.semanticscholar.org/paper/0834e74304b547c9354b6d7da6fa78ef47a48fa8.

[38] "A comprehensive survey of graph embedding: Problems, techniques, and applications," https://www.semanticscholar.org/paper/006906b6bbe5c1f378cde9fd86de1ce9e6b131da.

[39] "metapath2vec: Scalable representation learning for heterogeneous networks," https://www.semanticscholar.org/paper/c0af91371f426ff92117d2ccdadb2032bec23d2c.

[40] T. L. Scao, T. Wang, D. Hesslow, L. Saulnier, S. Bekman, S. Bari, S. R. Biderman, H. ElSahar, N. Muennighoff, J. Phang, O. Press, C. Raffel, V. Sanh, S. Shen, L. Sutawika, J. Tae, Z. X. Yong, J. Launay, and I. Beltagy, "What language model to train if you have one million gpu hours?" https://www.semanticscholar.org/paper/What-Language-Model-to-Train/bb15f3727f827a3cb88b5d3ca48415c09b40a88f.

[41] M. Mosbach, M. Andriushchenko, and D. Klakow, "On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines," https://www.semanticscholar.org/paper/On-the-Stability/8b9d77d5e52a70af37451d3db3d32781b83ea054.

[42] A. Matton and L. de Oliveira, "Emergent properties of finetuned language representation models," https://www.semanticscholar.org/paper/Emergent-Properties-of/79fdff5339017ec92b979efa4dff33d21a69b66e.

[43] S. Gehman, S. Gururangan, M. Sap, Y. Choi, and N. A. Smith, "Realtoxicityprompts: Evaluating neural toxic degeneration in language models," https://www.semanticscholar.org/paper/RealToxicityPrompts/399e7d8129c60818ee208f236c8dda17e876d21f.

[44] K. McGuffie and A. Newhouse, "The radicalization risks of gpt-3 and advanced neural language models," https://www.semanticscholar.org/paper/The-Radicalization/02fde8bfd9259a4f53316579eb0bf97213559e5c.

[45] A. Abid, M. S. Farooqi, and J. Y. Zou, "Persistent anti-muslim bias in large language models," https://www.semanticscholar.org/paper/Persistent-Anti-Muslim/4c2733d191e347753bb28afa46a1c55c65e085be.

[46] P. Indyk and R. Motwani, "Approximate nearest neighbors: towards removing the curse of dimensionality," in *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, 1998, pp. 604–613.

[47] E. Bernhardsson, "Annoy," https://github.com/spotify/annoy, 2013.

[48] Facebook Research, "Faiss," https://github.com/facebookresearch/faiss, 2019.

[49] S. Mizzaro, "Relevance: The whole history," *Journal of the American society for information science*, vol. 48, no. 9, pp. 810–832, 1997.

[50] P. Borlund, "The concept of relevance in IR," *Journal of the American Society for information Science and Technology*, vol. 54, no. 10, pp. 913–925, 2003.

[51] T. Saracevic, "The notion of relevance in information science," *Everybody knows what relevance is. But what is it really*, 2017.

[52] Y. Lu, Y. Dong, and L. Charlin, "Multi-XScience: A large-scale dataset for extreme multi-document summarization of scientific articles," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 8068–8074. [Online]. Available: https://aclanthology.org/2020.emnlp-main.648

[53] https://chat.openai.com/.

[54] K. W. Church and V. Kordoni, "Emerging trends: Sota-chasing," *Natural Language Engineering*, vol. 28, no. 2, pp. 249–269, 2022.

[55] R. Kinney, "Conference peer review with the semantic scholar api," https://blog.allenai.org/conference-peer-review-with-the-semantic-scholar-api-24ab9fce2324, 2021.

[56] M. Liberman and A. Prince, "On stress and linguistic rhythm," https://www.semanticscholar.org/paper/On-stress/b8ec853894551c0e7a822df50dc04eccd613d46f.

[57] J. A. Fisher, "The value of natural sounds," https://www.semanticscholar.org/paper/The-Value-of-Natural-Sounds-Fisher/f3044464aae6d74ef8b7c9a85810b5164e837378.

[58] F. V. Winnett, "Re-examining the foundations," https://www.semanticscholar.org/paper/Re-Examining-the-Foundations/52e0d57cccd3ffc9097ac9ee95c1a2214fdc1c7a.

[59] R. H. Pfeiffer, "Facts and faith in biblical history," https://www.semanticscholar.org/paper/Facts-and-Faith-in-Biblical-History-Pfeiffer/639c1e93818b157541bf5522a7cb2cf564119479.

[60] P. I. Nakov, A. S. Schwartz, M. Hearst *et al.*, "Citances: Citation sentences for semantic analysis of bioscience text," in *Proceedings of the SIGIR*, vol. 4. Citeseer, 2004, pp. 81–88.

[61] V. Qazvinian and D. Radev, "Identifying non-explicit citing sentences for citation-based summarization." in *Proceedings of the 48th annual meeting of the association for computational linguistics*, 2010, pp. 555–564.

[62] A. Abu-Jbara and D. Radev, "Reference scope identification in citing sentences," in *Proceedings of the 2012 conference of the North American Chapter of the Association for Computational Linguistics: Human language technologies*, 2012, pp. 80–90.

[63] N. Craswell, D. Hawking, and S. Robertson, "Effective site finding using link anchor information," in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 2001, pp. 250–257.

[64] A. Singh, M. D'Arcy, A. Cohan, D. Downey, and S. Feldman, "allenai/specter2," https://huggingface.co/allenai/specter2.

[65] M. Yasunaga, J. Leskovec, and P. Liang, "LinkBERT: Pretraining language models with document links," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 8003–8016. [Online]. Available: https://aclanthology.org/2022.acl-long.551

[66] M. Ostendorff, N. Rethmeier, I. Augenstein, B. Gipp, and G. Rehm, "Neighborhood contrastive learning for scientific document representations with citation embeddings," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec. 2022, pp. 11 670–11 688. [Online]. Available: https://aclanthology.org/2022.emnlp-main.802

[67] A. Cohan, S. Feldman, I. Beltagy, D. Downey, and D. S. Weld, "allenai/specter," https://huggingface.co/allenai/specter.

[68] "Globus," https://www.globus.org/.

[69] K. Church, "Better together team github," https://github.com/kwchurch/JSALT_Better_Together, 2023.

[70] G. Salton, A. Wong, and C.-S. Yang, "A vector space model for automatic indexing," *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[71] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information processing & management*, vol. 24, no. 5, pp. 513–523, 1988.

[72] T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," *Discourse processes*, vol. 25, no. 2-3, pp. 259–284, 1998.

[73] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

[74] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[75] "Nodevectors," https://github.com/VHRanger/nodevectors.

[76] A. Cohan, F. Dernoncourt, D. S. Kim, T. Bui, S. Kim, W. Chang, and N. Goharian, "A discourse-aware attention model for abstractive summarization of long documents," in

Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers). New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 615–621. [Online]. Available: https://aclanthology.org/N18-2097

[77] A. Fabbri, I. Li, T. She, S. Li, and D. Radev, "Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 1074–1084. [Online]. Available: https://aclanthology.org/P19-1102

[78] W. Li, X. Xiao, J. Liu, H. Wu, H. Wang, and J. Du, "Leveraging graph to improve abstractive multi-document summarization," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 6232–6243. [Online]. Available: https://aclanthology.org/2020.acl-main.555

[79] H. Hayashi, W. Kryscinski, B. McCann, N. Rajani, and C. Xiong, "What's new? summarizing contributions in scientific literature," in *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*. Dubrovnik, Croatia: Association for Computational Linguistics, May 2023, pp. 1019–1031. [Online]. Available: https://aclanthology.org/2023.eacl-main.72

[80] B.-J. Hsu, I. Shen, D. Eide, A. Chen, and R. Rogahn, "Microsoft academic graph," https://www.microsoft.com/en-us/research/project/microsoft-academic-graph/.

[81] "DOI foundation," https://www.doi.org/.

[82] "SciDocs - the dataset evaluation suite for specter," https://github.com/allenai/scidocs.

[83] A. Singh, M. D'Arcy, A. Cohan, D. Downey, and S. Feldman, "SciRepEval: A multi-format benchmark for scientific document representations," https://github.com/allenai/scirepeval.

[84] ——, "SciRepEval: A multi-format benchmark for scientific document representations," *ArXiv*, vol. abs/2211.13308, 2022.

[85] K. Wang, Z. Shen, C. Huang, C.-H. Wu, Y. Dong, and A. Kanakia, "Microsoft academic graph: When experts are not enough," *Quantitative Science Studies*, vol. 1, no. 1, pp. 396–413, 2020.

[86] "Pubmed," https://pubmed.ncbi.nlm.nih.gov/.

[87] "Pubmed central," https://www.ncbi.nlm.nih.gov/pmc/.

[88] "DBLP," https://dblp.org/.

[89] "arXiv," https://arxiv.org/.

[90] "ACL anthology," https://aclanthology.org/.

[91] "Social science research network," https://papers.ssrn.com/sol3/DisplayJournalBrowse.cfm.

[92] W. Hu, M. Fey, M. Zitnik, Y. Dong, H. Ren, B. Liu, M. Catasta, and J. Leskovec, "Open graph benchmark: Datasets for machine learning on graphs," *Advances in neural information processing systems*, vol. 33, pp. 22 118–22 133, 2020.

[93] W. Hu, M. Fey, H. Ren, M. Nakata, Y. Dong, and J. Leskovec, "OGB-LSC: A large-scale challenge for machine learning on graphs," *arXiv preprint arXiv:2103.09430*, 2021.

[94] J. Priem, H. A. Piwowar, and R. Orr, "OpenAlex: A fully-open index of scholarly works, authors, venues, institutions, and concepts," *ArXiv*, vol. abs/2205.01833, 2022.

[95] T. Strohman, W. B. Croft, and D. D. Jensen, "Recommending citations for academic papers," in *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2007.

[96] Q. He, J. Pei, D. Kifer, P. Mitra, and C. L. Giles, "Context-aware citation recommendation," in *The Web Conference*, 2010.

[97] Q. He, D. Kifer, J. Pei, P. Mitra, and C. L. Giles, "Citation recommendation without author supervision," in *Web Search and Data Mining*, 2011.

[98] S. Bethard and D. Jurafsky, "Who should I cite: learning literature search models from citation behavior," in *Proceedings of the 19th ACM international conference on Information and knowledge management*, 2010, pp. 609–618.

[99] https://jsalt2023.univ-lemans.fr/en/better-together-text-context.html, 2023.

[100] https://scholar.google.com/citations?user=E6aqGvYAAAAJ&hl=en.

[101] https://scholar.google.com/citations?user=zZhCPGkAAAAJ&hl=en.

[102] Q. Mei and K. Church, "Entropy of search logs: how hard is search? with personalization? with backoff?" in *Proceedings of the 2008 International Conference on Web Search and Data Mining*, 2008, pp. 45–54.

[103] A. S. Bacioiu, D. M. Sauntry, J. S. Boyle, L. C. W. Wong, P. F. Leonard, and R. Chandrasekar, "Method and apparatus for analysis and decomposition of classifier data anomalies," Sep. 16 2008, uS Patent 7,426,497.

[104] D. Belanger, K. Church, and A. Hume, "Virtual data warehousing, data publishing and call detail," in *Databases in Telecommunications: International Workshop, Co-located with VLDB-99, Edinburgh, Scotland, UK, September 6th, 1999. Proceedings 1.* Springer, 2000, pp. 106–117.

[105] K. W. Church and P. Hanks, "Word association norms, mutual information, and lexicography," *Computational Linguistics*, vol. 16, no. 1, pp. 22–29, 1990. [Online]. Available: https://www.aclweb.org/anthology/J90-1003

[106] O. Levy and Y. Goldberg, "Neural word embedding as implicit matrix factorization," in *NIPS*, 2014.

[107] https://research.northeastern.edu/about/institutes-centers/.

[108] J. Priem, H. Piwowar, and R. Orr, "OpenAlex: A fully-open index of scholarly works, authors, venues, institutions, and concepts," 2022. [Online]. Available: https://arxiv.org/abs/2205.01833