Building a Scientific Concept Hierarchy Database (SCHBASE)

Eytan Adar
University of Michigan
Ann Arbor, MI 48104
eadar@umich.edu

Srayan Datta
University of Michigan
Ann Arbor, MI 48104
srayand@umich.edu

Abstract

Extracted keyphrases can enhance numerous applications ranging from search to tracking the evolution of scientific discourse. We present SCHBASE, a hierarchical database of keyphrases extracted from large collections of scientific literature. SCHBASE relies on a tendency of scientists to generate new abbreviations that "extend" existing forms as a form of signaling novelty. We demonstrate how these keyphrases/concepts can be extracted, and their viability as a database in relation to existing collections. We further show how keyphrases can be placed into a semantically-meaningful "phylogenetic" structure and describe key features of this structure. The complete SCHBASE dataset is available at: http://cond.org/schbase.html.

1 Introduction

Due to the immense practical value to Information Retrieval and other text mining applications, keyphrase extraction has become an extremely popular topic of research. Extracted keyphrases, specifically those derived from scientific literature, support search tasks (Anick, 2003), classification and tagging (Medelyan et al., 2009), information extraction (Wu and Weld, 2008), and higher-level analysis such as the tracking of influence and dynamics of information propagation (Shi et al., 2010; Ohniwa et al., 2010). In our own work we use the extracted hierarchies to predict scientific emergence based on how rapidly new variants emerge. Keyphrases themselves capture a diverse set of scientific language (e.g., methods, techniques, materials, phenomena, processes, diseases, devices).

Keyphrases, and their uses, have been studied extensively (Gil-Leiva and Alonso-Arroyo, 2007). However, automated keyphrase *extraction* work has often focused on large-scale statistical techniques and ignored the scientific communication literature. This literature points to the complex ways in which keyphrases are created in light of competing demands: expressiveness, findability, succinct writing, signaling novelty, signaling community membership, and so on (Hartley and Kostoff, 2003; Ibrahim, 1989; Grange and Bloom, 2000; Gil-Leiva and Alonso-Arroyo, 2007). Furthermore, the tendency to extract keyphrases through statistical mechanisms often leads to flat keyphrase spaces that make analysis of evolution and emergence difficult.

Our contention, and the main motivation behind our work, is that we can do better by leveraging explicit mechanisms adopted by authors in keyphrase generation. Specifically, we focus on a tendency to expand keyphrases by adding terms, coupled with a pressure to abbreviate to retain succinctness. As we argue below, scientific communication has evolved the use of abbreviations to deal with various constraints. Abbreviations, and acronyms specifically, are relatively new in many scientific domains (Grange and Bloom, 2000; Fandrych, 2008) but are now ubiquitous (Ibrahim, 1989; Cheng, 2010).

Keyphrase selection is often motivated by increasing article findability within a domain (thereby increasing citation). This strategy leads to keyphrase reuse. A competing pressure, however, is to signal novelty in an author's work which is often done by creating new terminology (e.g., creating a "brand" around a system or idea). For

Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, pages 606–615,
Beijing, China, July 26-31, 2015.

example, a machine learning expert working on a new type of Support Vector Machine will want their article found when someone searches for "Support Vector Machine," but will also want to add their own unique brand. In response, they will often augment the original keyphrase (e.g., "Least-Squares Support Vector Machine") rather than inventing a completely new one. Unfortunately, continuous expansion will soon render a paper unreadable (e.g., one of many extensions to *Polymerase Chain Reaction* is *Standard Curve Quantitative Competitive Reverse Transcription Polymerase Chain Reaction*). Thus emerges a second strategy: abbreviation.

Our assertion is that abbreviations are a key mechanism for resolving competing demands. Authors can simultaneously expand keyphrases, thus maintaining both findability and novelty, while at the same time addressing the need to be succinct and non-repetitive. Of interest to us is the phenomena that if a new keyphrase expands an existing keyphrase that has an established abbreviation, the new keyphrase will also be abbreviated (e.g., LS-SVM and SVM). This tendency allows us to construct hierarchies of evolved keyphrases (rather than assuming a flat keyphrase space) which can be leveraged to identify emergence, keyphrase "mash-ups," and perform other high level analysis. As we demonstrate below, edges represent the rough semantic of EXTENDS or ISSUBTYPEOF. So if keyphrase *A* is connected to *B*, we can say *A* is a subtype of *B* (e.g., *A* is "Least-Squares Support Vector Machine" and *B* is "Support Vector Machine").

In this paper we introduce SCHBASE, a hierarchical database of keyphrases. We demonstrate how we can simply, but effectively, extract keyphrases by mining abbreviations from scientific literature and composing those keyphrases into semantically-meaningful hierarchies. We further show that abbreviations are a viable mechanism for building a domain-specific keyphrase database by comparing our extracted keyphrases to a number of author-defined and automatically-created keyphrase corpora. Finally, we illustrate how authors build upon each others' terminology over time to create new keyphrases.[1]

2 Related Work

Initial work in keyphrase extraction utilized heuristics that were based on the understood structure of scientific documents (Edmundson, 1969). As more data became available, it was possible to move away from heuristic cues and to leverage statistical techniques (Paice and Jones, 1993; Turney, 2000; Frank et al., 1999) that could identify keyphrases within, and between, documents. The guiding model in this approach is that phrases that appear as statistical "anomalies" (by some measure) are effective for summarizing a document or corpus. This style of keyphrase extraction represents much of the current state-of-the-art (Kim et al., 2010). Specific extensions in this space involve the use of network structures (Mihalcea and Tarau, 2004; Litvak and Last, 2008; Das Gollapalli and Caragea, 2014), part-of-speech features (Barker and Cornacchia, 2000; Hulth, 2003), or more sophisticated metrics (Tomokiyo and Hurst, 2003).

However, as we note above, these statistical approaches largely ignore the underlying tensions in scientific communication that lead to the creation of new keyphrases and how they are signaled to others. The result is that these techniques often find statistically "anomalous" phrases which often are not valid scientific concepts (but are simply uncommon phrasing), are unstructured and disconnected, and inflexible to size variance (as in the case of fixed length n-grams), and fail to capture extremely rare terminology.

The idea that abbreviations may be useful for keyphrase extraction has been partially realized. Nguyen et al., (2007) found that they could produce better keyphrases by extending existing models (Frank et al., 1999) to include an acronym indicator as a feature. That is, if a candidate phrase had an associated parenthetical acronym associated with it in the text a binary feature would be set. This approach has been implemented by others (Bordea and Buitelaar, 2010). We propose to expand on this idea by implementing a simple, but effective, solution by performing abbreviation extraction to build a hierarchical keyphrase database – a form of open-information extraction (Etzioni et al., 2008) on large scientific corpora.

3 Keyphrases and Hierarchies

Our high level strategy for finding an initial set of keyphrases is to mine a corpus for abbrevia-

[1] Full database available at: http://cond.org/schbase.html

tion expansions. This is a simple strategy, but as we show below, highly effective. Though the idea that abbreviations and keyphrases are linked fits within our understanding of scientific writing, we confirmed our intuition through a small experiment. Specifically, we looked at the 85 unique keyphrases (in this case, article titles) listed in the Wikipedia entry for *List of Machine Learning Concepts* (Wikipedia, 2014). These ranged from well known terms (e.g., *Support Vector Machines* and *Autoencoders*) to less known (e.g., *Information fuzzy networks*). In *all* 85 cases we were able to find an abbreviation on the Web (using Google) alongside the expansion (e.g., searching for the phrases *"Support Vector Machines (SVMs)"* or *"Information Fuzzy Networks (IFN)"*). Though there may be bias in the use of abbreviations in the Machine Learning literature, our experience has been that this holds in other domains as well. When a scientific keyphrase is used often enough, someone, somewhere, will have abbreviated it.

3.1 Abbreviation Extraction

To find all abbreviation expansions we use the unsupervised SaRAD algorithm (Adar, 2004). This algorithm is simple to implement, does not require extremely large amounts of data, works for both acronyms and more general abbreviations, and has been demonstrated as effective in various contexts (Adar, 2004; Schwartz and Hearst, 2003). However, our solution does not depend on a specific implementation, only that we are able to accurately identify abbreviation expansions.

Adar (2004) presents the full details for the algorithm, but for completeness we present the high level details. The algorithm progresses by identifying abbreviations inside of parentheses (defined as single words with at least one capital letter). The algorithm then extracts a "window" of text preceding the parenthesis, up to n words long (where n is the character length of the abbreviation plus padding). This window does not cross sentence boundaries. Within the window all possible "explanations" of the abbreviation are derived.

An explanation consists of a continuous subsequence of words that contain all the characters of the original abbreviation *in order*. For example, the window "determine the geographical distribution of ribonucleic acid" preceding the abbreviation "RNA" includes the explanations: "dete*r*mi*n*e the geogr*a*phical," "g*r*aphical distributio*n* of ribonucleic *a*cid" and "*r*ibo*n*ucleic *a*cid" (matching characters in italics). In the example above there are ten explanations (five unique). Each explanation is scored heuristically: 1 point for each abbreviation character at the start of a word; 1 point subtracted for every word between the explanation and the parenthesis; 1 point bonus if the explanation is adjacent to the parenthesis; 1 point subtracted for each extra word beyond the abbreviation length. For the explanations above, the scores are -4, 0, and 3 respectively. The highest scoring match (we require a minimum of 1 point) is returned as the mostly likely expansion.

In practice, pairs of extracted abbreviations/expansions are pulled from a large textual corpus. This both allows us to identify variants of expansions (e.g., different pluralization, spelling, hyphenation, etc.) as well as finding more plausible expansions (those that are repeated multiple times in a corpus). Thus, each expansion/abbreviation pair has an associated count which can be used to threshold and filter for increased quality. To discard units of measurement, single letter abbreviations and single word expansions are removed. We return to this decision later, but our experience is also that single word keyphrases are rare. Additionally, expansions containing brackets are not considered as they usually represent mathematical formulae.

3.1.1 The ABBREVCORPUS

In our experiments we utilize the ACM Digital Library (ACMDL) as our main corpus. Though the ACMDL is more limited than other collections, it has a number of desirable properties: spanning nearly the entire history (1954-2011) of a domain (Computer Science) with full-text and clean metadata. The corpus itself contains both journal and conference articles (77k and 197k, respectively).

In addition to the filtering rules described above, we manually constructed a set of filter terms to remove publication venues, agencies, and other institutions: 'university', 'conference', 'symposium', 'journal', 'foundation', 'consortium', 'agency', 'institute' and 'school' are discarded. We further normalize our keyphrases by lowercasing, removing hyphens, and using the Snowball stemmer (Porter, 2001) to merge plural variants. After stemming and normalizing, we found a total of 155,957 unique abbreviation expansions. Among these, 48,890 expansions occur more than once, 25,107 expansions thrice or more and 16,916 expansions four or more times. We refer to this collection as the AbbrevCorpus.

For each keyphrase we search within the full-text corpus to identify set of documents containing the keyphrase. This allowed us to find both the earliest mention of the keyphrase (the expansion, not the abbreviation) as well as overall popularity of keyphrases. We do not argue that abbreviations are the norm in the introduction of new keyphrases and may, in fact, only happen much later when the domain is familiar enough with the phrase.

To find the expansions in the full-text we utilize a modified suffix-tree that greedily finds the longest-matching phrase and avoids "double-counting". For example, if the text contains the phrase, "...we utilize a Least-Squares Support Vector Machine for ..." it will match against *Least-Squares Support Vector Machine* but not *Least Squares*, *Support Vector Machines*, or *Support Vector* (also keyphrases in our collection). The distribution of keyphrase frequency is a power-law (many keyphrases appearing once with a long tail) with exponent (α) of 2.17 (fit using Clauset et al., (2009)).

3.2 Building Keyphrase Hierarchies

We employ a very simple method of text containment to build keyphrase hierarchies from AbbrevCorpus. If a keyphrase A is a *substring* of keyphrase B, A is said to be contained by B ($B \rightarrow A$). If a third keyphrase, C, *contains* B and is *contained* by A, the containment link between A and B is dropped and two new ones ($A \rightarrow C$ and $C \rightarrow B$) are added. For example for the keyphrases, *circuit switching*, *optical circuit switching* and *dynamic optical circuit switching*, there are links from *optical circuit switching* to *circuit switching*, and *dynamic optical circuit switching* to *optical circuit switching*, but there is no link from *dynamic optical circuit switching* to *circuit switching*. The hierarchies formed in this manner are mostly trees, but in rare cases a keyphrase can have links to multiple branches. Example hierarchies are displayed in Figure 1.

For efficiency we sort all keyphrases by length (from largest to shortest) and iterate over each one, testing for containment in all previously "seen" keyphrases. This is computationally intensive, $O(n^2)$, but can be parallelized.

A potential issue with string containment is that negating prefixes can also appear (e.g., *non-monotonic reasoning* and *monotonic reasoning*). Our algorithm uses a dictionary of negations and can annotate the results. However, in practice we find that only .6% of our data has a leading negating-prefix ("internal" negating prefixes can also be caught in this way, but are similarly rare). It is an application-specific question if we want to consider such pairs as "siblings" or "parent-child" (with both supported).

4 Overlap with Keyphrase Corpora

To test our newly-constructed keyphrase database we generate a mixture of human- and machine-built datasets to compare. Our goal is to characterize both the intersection (keyphrases appearing in our corpus as well as the external datasets) as well as those keyphrases uniquely captured by each dataset.

4.1 ACM Author keyphrases (ACMCorpus)

The metadata for articles in ACM corpus contain author-provided keyphrases. In the corpus described above, we found 145,373 unique author-provided keyphrases after stemming and normalization. We discard 16,418 single-word keywords and those that do not appear in the full-text of any document. We retain 116,246 keyphrases which we refer to as the ACMCorpus.

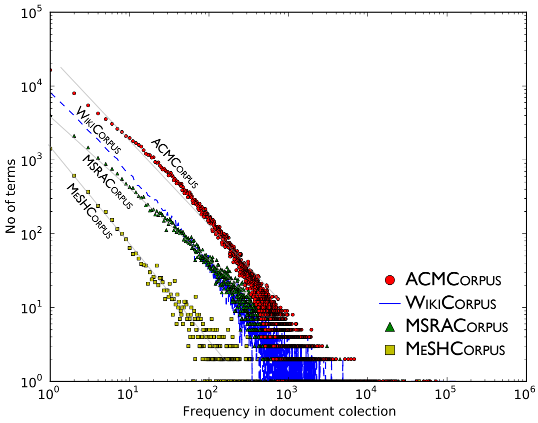

Figure 2: Keyphrase counts for the ACMCorpus (powerlaw $\alpha = 2.36$), WikiCorpus (2.49), MSRACorpus (2.55) and MeSHCorpus (2.7) within the ACM full-text.

4.2 Microsoft Academic (MSRACorpus)

Our second keyphrase dataset comes from the Microsoft Academic (MSRA) search corpus (Microsoft, 2015). While particularly focused on

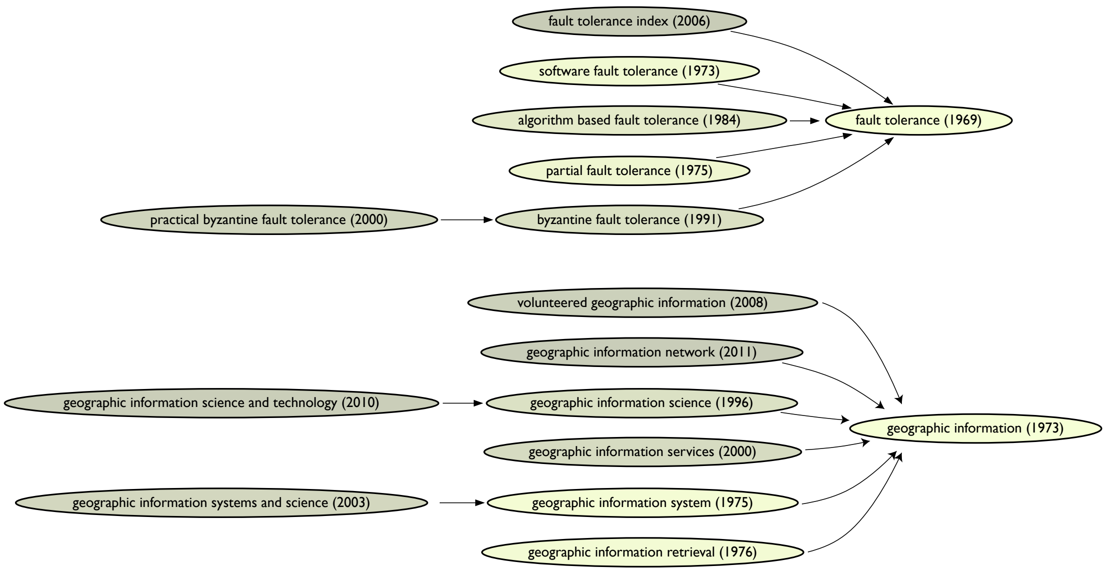
fault tolerance index (2006)
software fault tolerance (1973)
algorithm based fault tolerance (1984)
fault tolerance (1969)
partial fault tolerance (1975)
practical byzantine fault tolerance (2000)
byzantine fault tolerance (1991)
volunteered geographic information (2008)
geographic information network (2011)
geographic information science and technology (2010)
geographic information science (1996)
geographic information (1973)
geographic information services (2000)
geographic information systems and science (2003)
geographic information system (1975)
geographic information retrieval (1976)

Figure 1: Keyphrase hierarchy for *Fault Tolerance* (top) and *Geographic Information* (Bottom). Colors encode earliest appearance (brighter green is earlier)

Computer Science, this collection contains articles and keyphrases from over a dozen domains[2]. MSRA provides a list of keyphrases with unique IDs and different stemming variations of each keyphrase. There are a total of 46,978 (without counting stemming variations) of which 30,477 keyphrases occur in ACM full-text corpus after stemming and normalization (64% coverage).

4.3 MeSH (MESHCORPUS)

Medical Subject Headings (MeSH) (Lipscomb, 2000) is set of subject headings or descriptors in the life sciences domain. For the purpose of our work, we use the 27,149 keyphrases from the 2014 MeSH dataset. Similar to the other keyphrase lists we only use stemmed and normalized multi-word keywords that occur in in the ACM full-text corpus, which is 4,363 in case of MeSH.

4.4 Wikipedia (WIKICORPUS)

Scientific article headings in Wikipedia can often be used as a proxy for keyphrases. To collect relevant titles, we find Wikipedia articles that exactly match (in title name) existing MeSH and MSRA keyphrases. For these "seed" articles, we compile their categories and mark all the articles in these categories as potentially "relevant." However, as this also captures scientist names (e.g., a researcher's page may be placed under the "Computer Science" category), research institutes and other non-keyphrase matches, we use the page's infobox as a further filter. Pages containing "person," "place," infoboxes, in "book," "video game," "TV show" or other related "media" category, and those with geographical coordinates are removed. After applying these filters, we obtain 110,102 unique article titles (after stemming) which we treat as keyphrases. Of these, 39,974 occur in the ACM full-text corpus.

4.5 Results

The total overlap for ACMCORPUS, MESHCORPUS, MSRACORPUS and WIKICORPUS are 14.12%, 12.28%, 32.33% and 17.41% respectively. While these numbers seem low, it is worth noting that many of these terms only appear *once* in the ACM full-text corpus (see Figure 2).

Figure 3 illustrates the relationship between the number of times a keyphrase appears in the full-text and the probability that it will appear in ABBREVCORPUS. In all cases, the more often a keyphrase appears in the corpus, the more likely it is to have an abbreviation. If we qualitatively examine popular phrases that do not appear in ABBREVCORPUS we find mathematical forms (e.g., *of-the-form*, *well-defined* or *a priori*), and nouns/entities that are largely unrelated to scientific keyphrases (e.g., *New Jersey*, *Government Agency*, and *Private Sector*). More importantly,

[2] We know these keyphrases are algorithmically derived, but the details are not disclosed.

the majority of phrases that are never abbreviated are simply not Computer Science keyphrases (we return to this in Section 4.6).

We were somewhat surprised by the poor overlap of the ACMCorpus, even for terms that were very common in the full-text. We found that the cause was a large set of "bad" keyphrases. Specifically, 69.3k (69.5%) of author-defined keyphrases (occurring in ACMCorpus *but not* in AbbrevCorpus) are used as a keyword in only one paper. However, they appear more than once in the full-text – often many times. For example, one author (and only one) used *if and only if* as a keyphrase, which matched a great many articles. The result is that there is little correlation between the number of times a keyphrase appears in the full-text and how many times it used explicitly as a keyphrase in the document metadata. Because these will never be found as an abbreviation, they "pull" the mean probability down.

Instead of counting the number of times a keyphrase occurs in the full-text we generate a frequency count based on the number of times authors explicitly use it in the metadata. This new curve, labeled as ACMCorpus (key) in Figure 3 displays a very different tendency, with a rapid upward slope that peaks at 100% for frequently-occurring keyphrases. Notably, only 16k (16%) keyphrases appear once in full-text but are never abbreviated (far fewer than the 69.5% above).

It is worth briefly considering those terms that appear in AbbrevCorpus and not in the other keyphrases lists. We find roughly 17.6k, 24.7k, 19.4k, and 21.4k terms that appear in AbbrevCorpus (with a threshold of 2 to eliminate "noisy" expansions), but not in ACMCorpus, MeSHCorpus, MSRACorpus, and WikiCorpus respectively. As MeSH keyphrases tend to be focused on the biological keyphrases this is perhaps unsurprising but the high numbers for the author-provided ACM keyphrases is unexpected. We find that some of the keyphrases that are in AbbrevCorpus but not in ACMCorpus are highly specific (e.g., *Multi-object Evolutionary Algorithm Based on Decomposition* or *Stochastic Variable Graph Model*). However, many are also extremely generic terms that one *would* expect to find in a computer science corpus: *Run-Time Error Detection*, *Parallel Execution Tree*, and *Little Endian*. Our hypothesis is that these are often not the focus of a paper and are unlikely to be selected by the author. We believe this provides further evidence of the viability of the abbreviation approach to generating good keyphrase lists.

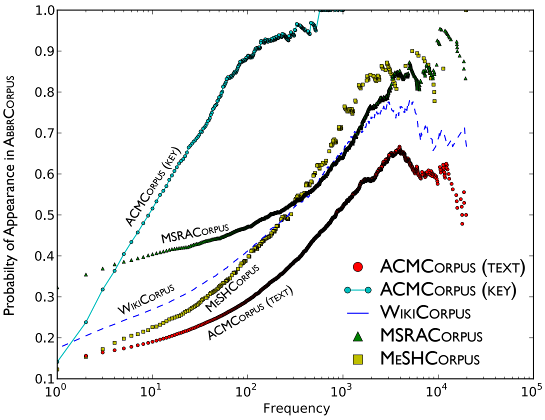

Figure 3: The probability of inclusion of keyphrases in AbbrevCorpus based on frequency of appearance in full text or, in the case if ACMCorpus (key), frequency of use as a keyword. At frequency x, the y value represents probability of appearence in AbbrevCorpus if we only consider terms that appear at least x times in the other corpus.

4.6 Domain keyphrases

When looking at keyphrases that appear in MeshCorpus but not in the AbbrevCorpus we find that many phrases do, in fact, appear in the full text but are never abbreviated. For example, *Color Perception* and *Blood Cell* both appear in ACM articles but are not abbreviated. Our hypothesis—which is motivated by the tendency of scientists to abbreviate terms that are deeply familiar to their community (Grange and Bloom, 2000)—is that terms that are possibly distant from the core domain focus tend not to be abbreviated. This is supported by the fact that these terms *are* abbreviated in other collections (e.g., one can find CP as an abbreviation for Color Perception in psychology and cognition work and BC, for Blood Cell, in medical and biological journals). Additional evidence is apparent in Figure 3 which shows that ACMCorpus keyphrases are more likely to be abbreviated (with far fewer repeats necessary). MSRACorpus, which contains many Computer Science articles, also has higher probabilities (though not nearly matching the ACM).

To test this systematically, we calculated semantic similarity between each keyphrase in the *WikiCorpus* dataset to "computer science." Specifically, we utilize Explicit Semantic Analysis (Gabrilovich and Markovitch, 2009) to calculate similarity. In this method, every segment of text is represented in a very high dimensional space in terms of keyphrases (based on Wikipedia categories). The similarity score for each term is between 0 (unrelated) and 1 (very similar).

Figure 4 demonstrates that with increasing similarity, the likelihood of abbreviation increases. From this, one may infer that to generate a domain-specific database that excludes unrelated keyphrases, the abbreviation-derived corpus is highly appropriate. Conversely, to get coverage of keyphrases from all scientific domains it is insufficient to mine for abbreviations in one specific domain's text. Even though a keyphrase may appear in the full-text it will simply never be abbreviated.

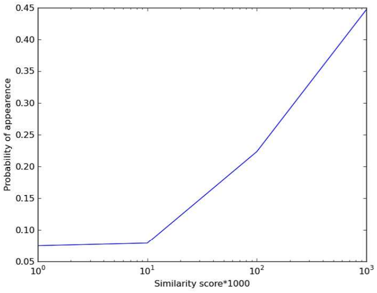

Figure 4: Probability of a keyphrase appearing in ABBREVCORPUS (y-axis) based on semantic similarity of the keyphrase to "Computer Science" (x-axis, binned exponentially for readability).

4.7 Keyphrase Hierarchies

Our hierarchy generation process (see Section 3.2) generated 1716 hierarchies accounting for 8661 unique keyphrases. Most of the hierarchies (1002 or 58%) only contained two nodes (a root and one child). The degree distribution, aggregated across all hierarchies, is again power-law ($\alpha = 2.895$). Hierarchy sizes are power-law distributed ($\alpha = 2.807$) and an average "diameter" (max height) of 1.135. The hierarchies contain a giant component with 2302 nodes and 2436 edges.

While most of our hierarchies are trees, keyphrases can connect to two independent branches. For example, *Least-Squares Support Vector Machines (LS-SVMs)* appears in both the *Least Squares* and *Support Vector* hierarchies. In total, 649 keyphrases appear in multiple hierarchies, the majority appearing 2. Only 17 keyphrases appear in 3 hierarchies. For example, the particularly long *Single Instruction Multiple Thread Evolution Strategy Pattern Search* appears in the *Evolution(ary) Strategy*, *Pattern Search*, and *Single-Instruction-Multiple-Thread* hierarchies. These collisions are interesting in that they reflect a mash-ups of different concepts, and by extension, different sub-disciplines or techniques. In some situations, where there is an overlap in many sub-keyphrases, this may indicate that two root keyphrases are in fact equivalent or highly related (e.g., *likelihood ratio* and *log likelihood*). We do not currently handle such ambiguity in SCHBASE.

To test the semantic interpretation of edges as EXTENDS/ISSUBTYPEOF we randomly sampled 200 edges and manually checked these. We found that in 92% (184) this interpretation was correct. The remaining 16 were largely an artifact of normalization errors rather than a wrong "type" (e.g., "session identifier" and "session id" where clearly a more accurate interpretation is ISEXPANSIONOF). We believe it is fair to say that the hierarchies we construct are the "skeleton" of a full EXTENDS hierarchy but one that is nonetheless fairly encompassing. Our qualitative analysis is that most keyphrases that share a type also share a root keyphrase (e.g., "classifier").

It is interesting to consider if edges which are derived by "containment" reflect a temporal pattern. That is, if keyphrase A EXTENDS B, does the first mention of A in the literature happen after B? We find that this is almost always the case. Among the 7136 edges generated by our algorithm only 165 (2.3%) are "reversed." Qualitatively, we find that these instances appear either due to missing data (the parent keyphrase first appeared outside the ACM) or publication ordering (in some cases the difference in first-appearance is only a year). In most situations the date is only 1-2 years apart. This high degree of consistency lends further support to the tendency of scientists to expand upon keyphrases over time.

Figure 5 depicts the mean change in length of "children" in keyphrase hierarchies. The numbers depicted are relative change. Thus, at year "0", the year the root keyphrase is introduced, there is no relative increase. Within 1 year, new children of that root are 50% larger in character length and after that children continue to "grow" as authors add additional keyphrases. A particularly obvious example of this is the branch for *Petri Net (PN)* which was extended as *Queueing Petri Net (QPN)* and then *Hierarchically Combined Queueing Petri Nets (HCQPN)* and finally *Extended Hierarchically Combined Queueing Petri Nets (EHCQPN)*. Notably, this may have implications to other extractors that assume fixed-sized entities over the history of the collection.

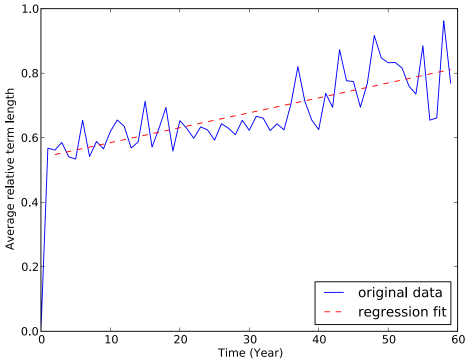

Figure 5: Average increase in character length of sub-keyphrases over time

5 Discussion and Future Work

Our decision to eliminate single-word keyphrases from consideration is an explicit one. Of the 145k keyphrases in the original ACMCORPUS (pre-filtering), 16,418 (11.29%) were single-word keyphrases. Our experience with the ACM author-defined keyphrases is that such terms are too generic to be useful as "scientific" keyphrases. For example, In all the ACM proceedings, the top-5 most common single-word keyphrases are *security*, *visualization*, *evaluation*, *design*, and *privacy*. Even in specific sub-domains, such as recommender systems (Proceedings of Recsys), the most popular single-word keyphrases are *personalization*, *recommendation*, *evaluation*, and *trust*. Contrast these to the most popular multi-word terms: *recommender system(s)*, *collaborative filtering*, *matrix factorization*, and *social network(s)*.

Notably, in the MSRA corpus, which is algorithmically filtered, only .46% (226 keyphrases) were single word. MeSH, in contrast, has a full 37% of keyphrases as single-term. In most situations these reflect chemical names (e.g., 382 single-word enzymes) or biological structures. In such a domain, and if these keyphrases are desirable, it may be advisable to retain single-word abbreviations. While it may seem surprising, even single words are often abbreviated (e.g., *Transaldolase* is "T" and *Ultrafiltration* is "U" or "U/F").

A second key observation is that while the ACM full-text corpus is large, it is by no means "big." We selected to use it because it controlled and "clean." However, we have also run our algorithms on the MSRA Corpus (which contains only abstracts) and CiteSeer (which contains full-text). Because the corpora contain more text we find significantly higher overlap with the different keyphrase corpora. However, this comes at the cost of not being able to isolate the domain-specific keyphrases. To put it differently, the broader full-text collections enable to us generate a more fleshed out keyphrase hierarchies that tracks keyphrases across all domains but which may not be appropriate for certain workloads.

Finally, it is worth considering the possibility of building hierarchies (and connecting them) by relations other than "containment." We have begun to utilize metrics such as co-occurrence of keyphrases (e.g., PMI) as well as higher level citation and co-citation structure in the corpora. Thus, we are able to connect terms that are highly related but are textually dissimilar. When experimenting with PMI, for example, we have found a diverse set of edge types including ISUSEDFOR (e.g., "n-gram language model" and "machine translation") or ISUSEDIN (e.g., "Expectation Maximization" and "Baum-Welch" or "euclidean algorithm" and "k-means"). By necessity, edges generated by this technique require an additional classification.

6 Summary

We have introduced SCHBASE, a simple, robust, and highly effective system and database of scientific concepts/keyphrases. By leveraging the incentive structure of scientists to expand existing ideas while simultaneously signaling novelty we are able to construct semantically-meaningful hierarchies of related keyphrases. The further tendency by authors to succinctly describe new keyphrases results in a general habit of utilizing abbreviations. We have demonstrated a mechanism to identify these keyphrases by extracting abbreviation expansions and have shown that these keyphrases cover the bulk of "useful" keyphrases within the domain of the corpus. We believe that SCHBASE will enable a number of applications ranging from search, categorization, and analysis of scientific communication patterns.

Acknowledgments

The authors thank the Microsoft Academic team, Jaime Teevan, Susan Dumais, and Carl Lagoze for providing us with data and advice. This work is supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20155. The U.S. government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

References

Eytan Adar. 2004. SaRAD: a simple and robust abbreviation dictionary. *Bioinformatics*, 20(4):527–533.

Peter Anick. 2003. Using terminological feedback for web search refinement: A log-based study. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, pages 88–95, New York, NY, USA. ACM.

Ken Barker and Nadia Cornacchia. 2000. Using noun phrase heads to extract document keyphrases. In Howard J. Hamilton, editor, *Advances in Artificial Intelligence*, volume 1822 of *Lecture Notes in Computer Science*, pages 40–52. Springer Berlin Heidelberg.

Georgeta Bordea and Paul Buitelaar. 2010. Deriunlp: A context based approach to automatic keyphrase extraction. In *Proceedings of the 5th international workshop on semantic evaluation*, pages 146–149. Association for Computational Linguistics.

Tsung O. Cheng. 2010. What's in a name? another unexplained acronym! *International Journal of Cardiology*, 144(2):291 – 292.

Aaron Clauset, Cosma Rohilla Shalizi, and Mark EJ Newman. 2009. Power-law distributions in empirical data. *SIAM Review*, 51(4):661–703.

Sujatha Das Gollapalli and Cornelia Caragea. 2014. Extracting keyphrases from research papers using citation networks. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*.

Harold P Edmundson. 1969. New methods in automatic extracting. *Journal of the ACM*, 16(2):264–285, April.

Oren Etzioni, Michele Banko, Stephen Soderland, and Daniel S. Weld. 2008. Open information extraction from the web. *Communications of the ACM*, 51(12):68–74, December.

Ingrid Fandrych. 2008. Submorphemic elements in the formation of acronyms, blends and clippings 147. *Lexis*, page 105.

Eibe Frank, Gordon W. Paynter, Ian H. Witten, Carl Gutwin, and Craig G. Nevill-Manning. 1999. Domain-specific keyphrase extraction. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'99, pages 668–673, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Evgeniy Gabrilovich and Shaul Markovitch. 2009. Wikipedia-based semantic interpretation for natural language processing. *Journal of Artificial Intelligence Research*, 34(1):443–498, March.

Isidoro Gil-Leiva and Adolfo Alonso-Arroyo. 2007. Keywords given by authors of scientific articles in database descriptors. *Journal of the American Society for Information Science and Technology*, 58(8):1175–1187.

Bob Grange and D.A. Bloom. 2000. Acronyms, abbreviations and initialisms. *BJU International*, 86(1):1–6.

James Hartley and Ronald N. Kostoff. 2003. How useful are 'key words' in scientific journals? *Journal of Information Science*, 29(5):433–438.

Anette Hulth. 2003. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, EMNLP '03, pages 216–223, Stroudsburg, PA, USA. Association for Computational Linguistics.

A.M. Ibrahim. 1989. Acronyms observed. *Professional Communication, IEEE Transactions on*, 32(1):27–28, Mar.

Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. 2010. Semeval-2010 task 5: Automatic keyphrase extraction from scientific articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 21–26. Association for Computational Linguistics.

Carolyn E. Lipscomb. 2000. Medical subject headings (mesh). *Bull Med Libr Assoc.* 88(3): 265266.

Marina Litvak and Mark Last. 2008. Graph-based keyword extraction for single-document summarization. In *Proceedings of the Workshop on Multi-source Multilingual Information Extraction and Summarization*, MMIES '08, pages 17–24, Stroudsburg, PA, USA. Association for Computational Linguistics.

Olena Medelyan, Eibe Frank, and Ian H. Witten. 2009. Human-competitive tagging using automatic keyphrase extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3 - Volume 3*, EMNLP '09, pages 1318–1327, Stroudsburg, PA, USA. Association for Computational Linguistics.

Microsoft. 2015. Microsoft academic search. http://academic.research.microsoft.com. Accessed: 2015-2-26.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into texts. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 404–411, Barcelona, Spain, July. Association for Computational Linguistics.

ThuyDung Nguyen and Min-Yen Kan. 2007. Keyphrase extraction in scientific publications. In Dion Hoe-Lian Goh, Tru Hoang Cao, Ingeborg Torvik Sølvberg, and Edie Rasmussen, editors, *Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers*, volume 4822 of *Lecture Notes in Computer Science*, pages 317–326. Springer Berlin Heidelberg.

Ryosuke L. Ohniwa, Aiko Hibino, and Kunio Takeyasu. 2010. Trends in research foci in life science fields over the last 30 years monitored by emerging topics. *Scientometrics*, 85(1):111–127.

Chris D. Paice and Paul A. Jones. 1993. The identification of important concepts in highly structured technical papers. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '93, pages 69–78, New York, NY, USA. ACM.

Martin F. Porter. 2001. Snowball: A language for stemming algorithms. http://snowball.tartarus.org/texts/introduction.html. Accessed: 2015-2-26.

Ariel S Schwartz and Marti A Hearst. 2003. A simple algorithm for identifying abbreviation definitions in biomedical text. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, page 451462.

Xiaolin Shi, Jure Leskovec, and Daniel A. McFarland. 2010. Citing for high impact. In *Proceedings of the 10th Annual Joint Conference on Digital Libraries*, JCDL '10, pages 49–58, New York, NY, USA. ACM.

Takashi Tomokiyo and Matthew Hurst. 2003. A language model approach to keyphrase extraction. In *Proceedings of the ACL 2003 Workshop on Multiword Expressions: Analysis, Acquisition and Treatment - Volume 18*, MWE '03, pages 33–40, Stroudsburg, PA, USA. Association for Computational Linguistics.

Peter D. Turney. 2000. Learning algorithms for keyphrase extraction. *Information Retrieval*, 2(4):303–336, May.

Wikipedia. 2014. Wikipedia: List of machine learning concepts. http://en.wikipedia.org/wiki/List_of_machine_learning_concepts. Accessed: 2015-2-26.

Fei Wu and Daniel S. Weld. 2008. Automatically refining the wikipedia infobox ontology. In *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, pages 635–644, New York, NY, USA. ACM.