# Modeling Concept Dependencies in a Scientific Corpus

**Jonathan Gordon**, **Linhong Zhu**, **Aram Galstyan**, **Prem Natarajan**, and **Gully Burns**
USC Information Sciences Institute
Marina del Rey, CA, USA
{jgordon, linhong, galstyan, pnataraj, burns}@isi.edu

## Abstract

Our goal is to generate reading lists for students that help them optimally learn technical material. Existing retrieval algorithms return items directly relevant to a query but do not return results to help users read about the concepts supporting their query. This is because the dependency structure of concepts that must be understood before reading material pertaining to a given query is never considered. Here we formulate an information-theoretic view of concept dependency and present methods to construct a "concept graph" automatically from a text corpus. We perform the first human evaluation of concept dependency edges (to be published as open data), and the results verify the feasibility of automatic approaches for inferring concepts and their dependency relations. This result can support search capabilities that may be tuned to help users learn a subject rather than retrieve documents based on a single query.

## 1 Introduction

Corpora of technical documents, such as the ACL Anthology, are valuable for learners, but it can be difficult to find the most appropriate documents to read in order to learn about a concept. This problem is made more complicated by the need to trace the ideas back to those that need to be learned first (e.g., before you can learn about Markov logic networks, you should understand first-order logic and probability). That is, a crucial question when learning a new subject is "What do I need to know before I start reading about this?"

To answer this question, learners typically rely on the guidance of domain experts, who can devise pedagogically valuable reading lists that order documents to progress from prerequisite to target concepts. Thus, it is desirable to have a model where each concept is linked to the prerequisite concepts it depends upon – a *concept graph*. A manually constructed concept graph excerpt related to automatic speech recognition is shown in Figure 1. The dependency relation between two concepts is interpreted as whether understanding one concept would help a learner understand the other.

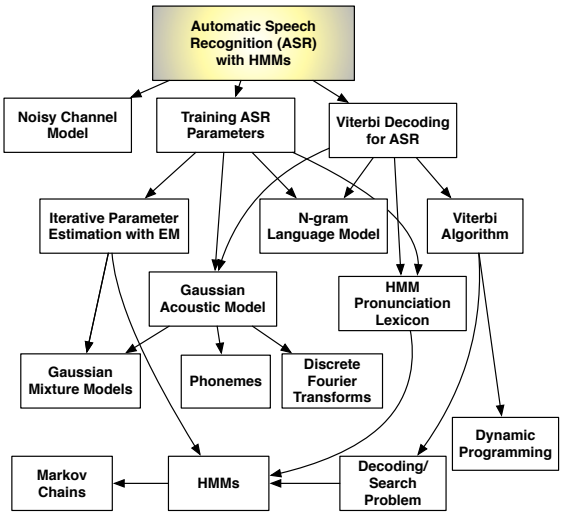


Figure 1: A human-authored concept graph excerpt, showing possible concepts related to automatic speech recognition and their concept dependencies.

Representing a scientific corpus in this way can improve tasks such as curriculum planning (Yang et al., 2015), automatic reading list generation (Jardine, 2014), and improving education quality (Rouly et al., 2015). Motivated by the importance of representing the content of a scientific corpus as a concept graph, the challenge we address in this work is to automatically infer the concepts and their dependency relations.

Towards this end, we first instantiate each concept as a topic from statistical topic modeling (Blei et al., 2003). To link concepts with directed depen-

dency edges, we propose the use of information-theoretic measures, which we compare against baseline methods of computing word similarity, hierarchical clustering, and citation prediction. We then gather human annotations of concept graph nodes and edges learned from the ACL Anthology, which we use to evaluate these methods.

The main contributions of this paper are:

1 We introduce the concept graph representation for modeling the technical concepts in a corpus and their relations.
2 We present information-theoretic approaches to infer concept dependence relations.
3 We perform the first human annotation of concept dependence for a technical corpus.
4 We release the human annotation data for use in future research.

In the following section, we contrast this problem with previous work. We then describe the concept graph framework (Section 3) and present automatic approaches for inferring concept graphs (Section 4). The details of human evaluation are presented in Section 5. We discuss some interesting open questions related to this work in Section 6 before concluding this work.

## 2 Related Work

There is a long history of work on identifying structure in the contents of a text corpus. Our approach is to link documents to concepts and to model relations among these concepts rather than to identify the specific claims (Schäfer et al., 2011) or empirical results (Choi et al., 2016) in each document. In this section, we first provide an overview of different relations between concepts, followed by discussion of some representative methods for inferring them. We briefly discuss the differences between these relations and the concept dependency relation we are interested in.

**Similarity** Concepts are similar to the extent that they share content. Grefenstette (1994) applied the Jaccard similarity measure to relate concepts to each other. White and Jose (2004) empirically studied 10 similarity metrics on a small sample of 10 pairs of topics, and the results suggested that correlation-based measures best match general subject perceptions of search topic similarity.

**Hierarchy** Previous work on linking concepts has usually been concerned with forming subsumption hierarchies from text (Woods, 1997; Sanderson and Croft, 1999; Cimiano et al., 2005) – e.g., *Machine translation* is part of *Natural language processing* – and more recent work does so for statistical topic models. Jonyer et al. (2002) applied graph-based hierarchical clustering to learn hierarchies from both structured and unstructured data. Ho et al. (2012) learn a topic taxonomy from the ACL Anthology and from Wikipedia with a method that scales linearly with the number of topics and the tree depth.

**Other relations** Every pair of concepts is statistically correlated with each other based on word co-occurrence (Blei and Lafferty, 2006) providing a simple baseline metric for comparison. For a topic modeling approach performed over document citation links rather than over words or n-grams, Wang et al. (2013) gave a topic $A$'s dependence on another topic $B$ as the probability of a document in $A$ citing a document in $B$.

Our approach to studying *concept dependence* differs from the relations derived from similarity, hierarchy, correlation and citation mentioned above, but intuitively they are related. We thus adapt one representative method for the similarity (Grefenstette, 1994), hierarchy (Jonyer et al., 2002), and citation likelihood (Wang et al., 2013) relations as baselines for computing concept dependency relations in Section 4.2.3.

Concept dependence is also related to curriculum planning. Yang et al. (2015) and Talukdar and Cohen (2012) studied prerequisite relationships between course material documents based on external information from Wikipedia. They assumed that hyperlinks between Wikipedia pages and course material indicate a prerequisite relationship. With this assumption, Talukdar and Cohen (2012) use crowdsourcing approaches to obtain a subset of the prerequisite structure and train a maximum entropy–based classifier to identify the prerequisite structure. Yang et al. (2015) applied both classification and learning to rank approaches in order to classify or rank prerequisite structure.

## 3 Concept Graph Representation of a Text Corpus

We represent the scientific literature as a labeled graph, where nodes represent both documents and concepts – and, optionally, metadata (such as author, title, conference, year) and features (such as

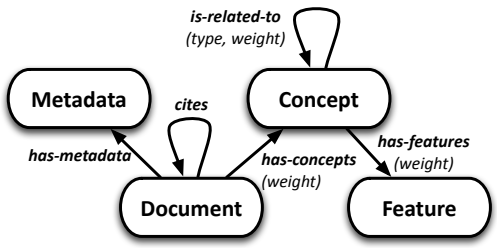


Figure 2: The Concept Graph Data Schema. Each node is a class and edges are named relations between classes (with associated attributes).

words, or n-grams) – and labeled edges represent the relations between nodes. Figure 2 shows an example schema for a concept graph representation for a scientific corpus.

*Concepts* are abstract and require a concrete representation. In this work, we use statistical topic modeling, where each topic – a multinomial distribution over a vocabulary of words – is taken as a single concept. Documents are linked to concepts by weighted edges, which can be derived from the topic model's document–topic composition distributions. Other approaches to identifying concepts are considered in Section 6.

Concepts exhibit various relations to other concepts, such as *hierarchy*, connecting more general and more specific concepts; *similarity*; and *correlation*. We model each concept as a node and concept-to-concept relations as directed, weighted, labeled edges. The label of an edge denotes the type of relation, such as "is similar to", "depends on", and "relates to", and the weights represent the strength of different relations.

In this work, we focus on *concept dependency*, which is the least studied of these relations and, intuitively, the most important for learners. We consider there to be a dependency relation between two concepts if understanding one concept would help you to understand the other. This notion forms the core of our human-annotated data set which demonstrates that this idea is meaningful and robust for expert annotators when asked to judge if there exists a dependency relation between two concepts defined by LDA topics (see Section 5.2).

## 4 Learning the Concept Graph

### 4.1 Identifying Concepts

The representation of concepts using topics is very general, and any effective topic modeling approach can be applied. These include probabilistic latent semantic indexing (PLSI) (Hofmann, 1999), latent Dirichlet allocation (LDA) (Blei et al., 2003), and non-negative matrix factorization (NMF) (Arora et al., 2012). In our experiments, we use the open-source tool Mallet (McCallum, 2002), which provides a highly scalable implementation of LDA; see Section 5.1 for more details.

### 4.2 Discovering Concept Dependency Relations

Identifying concept dependency relations between topics is the key step for building a useful concept graph. These relations add semantic structure to the contents of the text corpus, and they facilitate search and ordering in information retrieval. In this section, as a proof-of-concept, we propose two information-theoretic approaches to learn concept dependency relations: an approach based on cross entropy and another based on information flow.

#### 4.2.1 Cross-entropy Approach

The intuition of the cross-entropy approach is simple: Given concepts $c_i$ and $c_j$, if most of the instances of $c_i$ can be explained by the occurrences of $c_j$, but not vice versa, it is likely that $c_i$ depends on $c_j$. For example, if $c_i$ is *Markov logic networks* (*MLNs*) and $c_j$ is *Probability*, we might say that observing *MLNs* depends on seeing *Probability* since most of the times that we see *MLNs*, we also see *Probability*, but the opposite does not hold.

Given concepts $c_i$ and $c_j$, the cross-entropy approach predicts that $c_i$ depends on $c_j$ if they satisfy these conditions:

1 The distribution of $c_i$ is better approximated by that of $c_j$ than the distribution of $c_j$ is approximated by that of $c_i$.
2 The co-occurrence frequency of instances of $c_i$ and $c_j$ is relatively higher than that of a non-dependency pair.

Therefore, to predict the concept dependency relation, we need to examine whether the distribution of $c_i$ could well approximate the distribution of $c_j$ and the joint distribution of $c_i$ and $c_j$. For this, we use cross entropy and joint entropy:

**Cross entropy** measures the difference between two distributions. Specifically, the cross entropy for the distributions $X$ and $Y$ over a given set is defined as:

$$H(X;Y) = H(X) + D_{KL}(X||Y) \qquad (1)$$

where $H(X)$ is the entropy of $X$, and $D_{KL}(X||Y)$ is the Kullback–Leibler divergence of an estimated distribution $Y$ from true distribution $X$. Therefore, $H(X;Y)$ examines how well the distribution of $Y$ approximates that of $X$.

**Joint entropy** measures the information we obtained when we observe both $X$ and $Y$. The joint Shannon entropy of two variables $X$ and $Y$ is defined as:

$$H(X,Y) = \sum_X \sum_Y P(X,Y) \log_2 P(X,Y) \quad (2)$$

where $P(X,Y)$ is the joint probability of these values occurring together.

Based on the conditions listed above and these definitions, we say that $c_i$ depends on $c_j$ if and only if they satisfy the following constraints:

$$\begin{aligned} H(c_i;c_j) &> H(c_j;c_i) \\ H(c_i,c_j) &\leq \theta \end{aligned} \quad (3)$$

with $\theta$ as a threshold value, which can be interpreted as "the average joint entropy of any non-dependence concepts". The weight of the dependency is defined as:

$$D_{\mathrm{CE}}(c_i,c_j) = H(c_i;c_j)$$

The cross-entropy method is general and can be applied to different distributions used to model concepts, such as distributions of relevant words, of relevant documents, or of the documents that are cited by relevant documents.

#### 4.2.2 Information-flow Approach

Now we consider predicting concept dependency relations from the perspective of navigating information. Imagine that we already have a perfect concept dependency graph. When we are at a concept node (e.g., reading a document about it), the navigation is more likely to continue to a concept it depends on than to other concepts that it doesn't depend on. To give a concrete example, if we are navigating from the concept *Page rank*, it is more likely for us to jump to *Eigenvalue* than to *Language model*. Therefore, if concept $c_i$ depends on concept $c_j$, then $c_j$ generally receives more navigation hits than $c_i$ and has higher "information flow".

Based on this intuition, we can predict concept dependency relations using information flow: Given concepts $c_i$ and $c_j$, $c_i$ depends on $c_j$ if they satisfy these conditions:

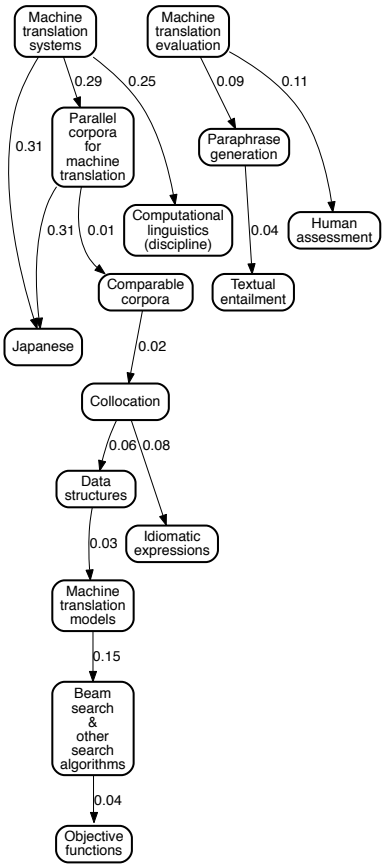


Figure 3: A concept graph excerpt related to machine translation, where concepts are linked based on cross entropy. Concepts are represented by manually chosen names, and links to documents are omitted.

1 The concept $c_i$ receives relatively lower navigation hits than $c_j$.
2 The number of navigation traces from concept $c_i$ to $c_j$ is much stronger than that to another non-dependent concept $c_k$.

While we do not have data for human navigation between concepts, a natural way to simulate this is through information flow. As proposed by Rosvall and Bergstrom (2008), we use the probability flow of random walks on a network as a proxy for information flow in the real system. Given any observed graph $G$, the information score $I(v)$ of a node $v$, is defined as its steady state visit frequency. The information flow $I(u,v)$ from node $u$ to node $v$, is consequently defined as the transition probability (or "exit probability") from $u$ to $v$.

To this end, we construct a graph connecting concepts by their co-occurrences in documents, and we can use either Map Equation (Rosvall and

Bergstrom, 2008) or Content Map Equation (Smith et al., 2014) to compute the information flow network and the information score for each concept node. The details are outlined as follows:

1 Construct a concept graph $G^{co}$ based on co-occurrence observations. We define weighted, undirected edges within the concept graph based on the number of documents in which the concepts co-occur. Formally, given concepts $c_i$ and $c_j$ and a threshold $0 \leq \tau \leq 1$, the weighted edge is calculated as:

$$w^{co}(c_i, c_j) = \begin{cases} \sum_d p(c_i|d)p(c_j|d) & \text{if } p(c|d) > \tau \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

2 Given the graph $G^{co}$, we compute the information score $I(c)$ for each concept node $c$ and information flow $I(c_i, c_j)$ between a pair of nodes $c_i$ and $c_j$. For the details of calculating $I(c)$ and $I(c_i, c_j)$, refer to Map Equation (Rosvall and Bergstrom, 2008) and Content Map Equation (Smith et al., 2014).

3 Given two concepts $c_i$ and $c_j$, we link $c_i$ to $c_j$ with a directed edge if $I(c_i) > I(c_j)$ with weight:

$$D_{\text{IF}}(c_i, c_j) = I(c_i, c_j)$$

The information flow approach for inferring dependency can be further improved with a few true human navigation traces. As introduced earlier, the concept graph representation facilitates applications such as reading list generation, and document retrieval. Those applications enable the collection of human navigation traces, which can provide a better approximation of dependency relation.

#### 4.2.3 Baseline Approaches

**Similarity Relations** Intuitively, concepts that are more similar (e.g., *Machine translation* and *Machine translation evaluation*) are more likely to be connected by concept dependency relations than less similar concepts are. As a baseline, we compute the Jaccard similarity coefficient based on the top 20 words or n-grams in the concept's topic word distributions.

**Hierarchical Relations** Previous work has looked at learning hierarchies that connect broader topics (acting as equivalent proxies for concepts in our work) to more specific subtopics (Cimiano et al., 2005; Sanderson and Croft, 1999). We compare against a method for doing so to see how close identifying hierarchical relations comes to our goal of identifying concept dependency relations. Specifically, we perform agglomerative clustering over the topic–topic co-occurrence graph $G^{co}$ with weights defined in Eq. 4, in order to obtain the hierarchical representation for concepts.

**Citation-based** Given concepts $c_i$ and $c_j$, if the documents that are highly related to $c_j$ are cited by most of the instances of $c_i$, $c_i$ may depend on $c_j$. Wang et al. (2013) used this approach in the context of CitationLDA topic modeling, where topics are learned from citation links rather than text. We adapt this for regular LDA so that the concept $c_i$ depends on $c_j$ with weight

$$\text{D}_{\text{Cite}}(c_i, c_j) = \sum_{d_1 \in D} \sum_{d_2 \in C_{d_1}} T_{1,i} T_{2,j} \quad (5)$$

where $D$ is the set of all documents, $C_d$ are the documents cited by $d$, and $T_{x,y}$ is the distribution of documents $d_x$ composed of concepts $c_y$. For this method, we return a score of 0 if the concepts do not co-occur in at least three documents.

## 5 Evaluation of Concept Graphs

There are two main approaches to evaluating a concept graph: We can directly evaluate the graph, using human judgments to measure the quality of the concepts and the reliability of the links between them. Alternatively, we can evaluate the application of a concept graph to a task, such as ordering documents for a reading list or recommending documents to cite when writing a paper.

Our motivation to build a concept graph from a technical corpus is to improve performance at the task of reading list generation. However, an applied evaluation makes it harder to judge the quality of the concept graph itself. Each document contains a combination of concepts, which have different ordering restrictions, and other factors also affect the quality of a reading list, such as the classification of document difficulty and type (e.g., survey, tutorial, or experimental results). As such, we focus on a direct human evaluation of our proposed methods for building a concept graph and leave the measure of applied performance to future work.

### 5.1 Corpus and its Evaluation Concept Graphs

For this evaluation, the scientific corpus we use is the ACL Anthology. This consists of articles published in a variety of journals, conferences,

and workshops related to computational linguistics. Specifically, we use a modified copy of the plain text distributed for the ACL Anthology Network (AAN), release 2013 (Radev et al., 2013), which includes 23,261 documents from 1965 to 2013. The AAN includes plain text for documents, with OCR performed using PDFBox. We manually substituted OmniPage OCR output from the ACL Anthology Reference Corpus, version 1 (Bird et al., 2008) for documents where it was observed to be of higher quality. The text was processed to join words that were split across lines with hyphens. We manually removed documents that were not written in English or where text extraction failed, leaving 20,264 documents, though this filtering was not exhaustive.

The topic model we used was built using the Mallet (McCallum, 2002) implementation of LDA. It is composed of bigrams, filtered of typical English stop words before the generation of bigrams, so that, e.g., "word to word" yields the bigram "word word". We generated topic models consisting of between 20 and 400 topics and selected a 300-topic model based on manual inspection. Documents were linked to concepts based on the document's LDA topic composition. The concept nodes for each topic were linked in concept dependency relations using each of the methods described in Section 4, producing five concept graphs to evaluate. We applied the general cross-entropy method to the distribution of top-$k$ bigrams for each concept. For all methods, the results we report are for $k = 20$. Changing this value shifts the precision–recall trade-off, but in our experiments, the relative performance of the methods are generally consistent for different values of $k$.

Since it is impractical to manually annotate all pairs of concept nodes from a 300-node graph, we selected a subset of edges for evaluation. Intuitively, the evaluation set should satisfy the following sampling criteria: (1) The evaluation set should cover the top weighted edges for a precision evaluation. (2) The evaluation set should cover the bottom-weighted edges for a recall evaluation. (3) The evaluation set should provide low-biased sampling. With respect to these requirements, we generated an evaluation edge set as the union of the following three sets:

1 Top-20 edges for each approach (including baseline approaches)
2 A random shuffle selection from the union of the top-50 and bottom-50 edges in terms of the baseline word similarity.[1]
3 A random shuffle section from the union of top-100 edges in terms of the proposed approaches.

| Judges | All | Coherent | Related | Dependent |
|---|---|---|---|---|
| Non-NLP | 0.407 | 0.446 | 0.305 | 0.329 |
| NLP | 0.526 | 0.610 | 0.448 | 0.395 |
| All | 0.467 | 0.529 | 0.354 | 0.357 |

Table 1: Inter-annotator agreement measured as Pearson correlation.

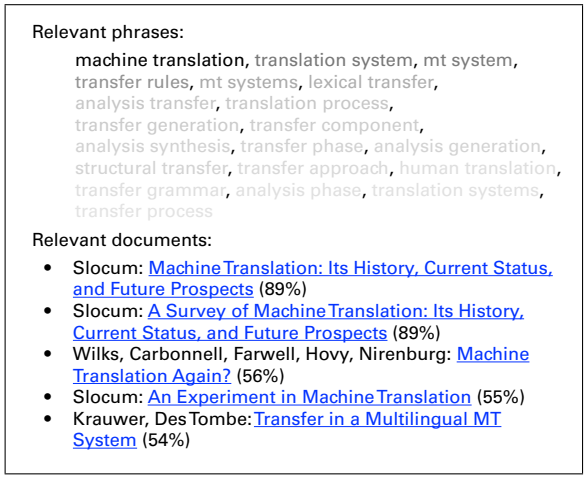


Figure 4: An example of the presentation of a topic for human evaluation.

### 5.2 Human Annotation

For annotation, we present pairs of topics followed by questions. Each topic is presented to a judge as a list of the most relevant bigrams in descending order of their topic-specific "collapsed" probabilities. These are presented in greyscale so that the most relevant items appear black, fading through grey to white as the strength of that item's association with the topic decreases. The evaluation interface also lists the documents that are most relevant to the topic, linked to the original PDFs. These documents can be used to clarify the occurrence of unfamiliar terms, such as author names or common examples that may show up in the topic representation. An example topic is shown in Figure 4.

For each topic, judges were asked:

1 How clear and coherent is Topic 1?
2 How clear and coherent is Topic 2?

[1] We observe that usually if the edge strength in terms of one of the information-theoretic methods is zero, the word similarity is zero as well, but if the word similarity is zero, the edge strength in terms of the proposed methods may be non-zero.

| Edges | Top 20 | Top 150 | | | All scores $> 0$ | | |
|---|---|---|---|---|---|---|---|
| | Prec. | Prec. | Rec. | $f_1$ | Prec. | Rec. | $f_1$ |
| Cross entropy ($D_{CE}$) | **0.851** | 0.765 | 0.358 | 0.487 | 0.693 | 0.670 | 0.681 |
| Information flow ($D_{IF}$) | 0.793 | 0.696 | 0.311 | 0.429 | 0.693 | 0.323 | 0.441 |
| Word similarity ($D_{Sim}$) | 0.808 | **0.768** | 0.382 | **0.511** | **0.768** | 0.382 | 0.511 |
| Hierarchy ($D_{Hier}$) | 0.680 | 0.692 | 0.297 | 0.416 | 0.686 | 0.638 | 0.661 |
| Cite ($D_{Cite}$) | 0.693 | 0.718 | 0.343 | 0.465 | 0.693 | 0.670 | 0.681 |
| Random | 0.659 | 0.661 | **0.580** | 0.500 | 0.658 | **1.000** | **0.794** |

Table 2: Precision, recall, and f-scores (with different thresholds for which edges are included) for the methods of predicting dependency relations between concepts described in Section 4.2.

*If both topics are at least somewhat clear:*

3 How related are these topics?
4 Would understanding Topic 1 help you to understand Topic 2?
5 Would understanding Topic 2 help you to understand Topic 1?

For each question, they could answer "I don't know" or select from an ordinal scale:

1 Not at all
2 Somewhat
3 Very much

The evaluation was completed by eight judges with varying levels of familiarity with the technical domain. Four judges are NLP researchers: Three PhD students working in the area and one of the authors. Four judges are familiar with NLP but have less experience with NLP research: two MS students, an AI PhD student, and one of the authors. The full evaluation was divided into 10 sets taking a total of around 6–8 hours per person to annotate. Their overall inter-annotator agreement and the agreement for each question type is given in Table 1. Agreement is higher when we consider only judgments from NLP researchers, but in all cases is moderate, indicating the difficulty of interpreting statistical topics as concepts and judging the strength (if any) of the concept dependency relation between them.

The topic coherence judgments that were collected served to make each human judge consider how well she understood each topic before judging their dependence. The topic relatedness questions provided an opportunity to indicate that if the annotator recognized a relation between the topics without needing to say that their was a dependence.

### 5.3 Evaluation of Automatic Methods

To measure the quality of the concept dependency edges in our graphs, we compute the average precision for the strongest edges in each concept graph, up to three thresholds: the top 20 edges, the top 150, and all edges with strength $> 0$. These precision scores are in Table 2 as well as the corresponding recall, and $f_1$ scores for the larger thresholds. Despite the difference in inter-annotator agreement reported in Table 1, the ordering of methods by precision is the same whether we consider only the judgments of NLP experts, non-NLP judges, or everyone, so we only report the average across all annotators.

When we examine the results of precision at 20 – the strongest edges predicted by each method – we see that the cross-entropy method performs best. For comparison, we report the accuracy of a baseline of random numbers between 0 and 1. While all methods have better than chance precision, the random baseline has higher recall since it predicts a dependency relation of non-zero strength for all pairs. As we consider edges predicted with lower confidence, the word similarity approach shows the highest precision. A limitation of the word similarity baseline is that it is symmetric while concept dependence relations can be asymmetric.

Annotators marked many pairs of concepts as being at least somewhat co-dependent. E.g., understanding *Speech recognition* strongly helps you understand *Natural language processing*, but being familiar with this broader topic also somewhat helps you understand the narrower one. The precision scores we report count both annotations of concept dependence ("Somewhat" and "Very much") as positive predictions, but other evaluation metrics might show a greater benefit for methods like $D_{CE}$ that can predict dependency with asymmetric strengths.

## 6 Discussion

Another natural evaluation of an automatically generated concept graph would be to compare it to a

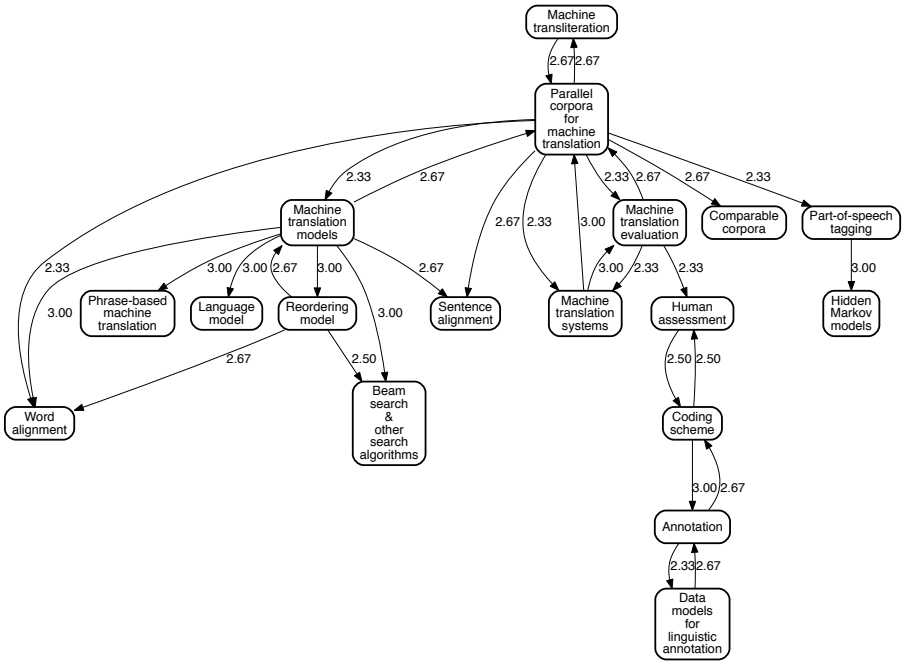


Figure 5: A concept graph excerpt related to machine translation, where concepts are joined based on the judgments of human annotators. Concepts are represented by manually chosen names, and links to documents are omitted.

human-generated gold standard, where an expert has created concept nodes at the optimal level of generality and linked these by her understanding of the conceptual dependencies among concepts in the domain. However, there are several difficulties with this approach: (1) It is quite labor-intensive to manually generate a concept graph; (2) we expect only moderate agreement between graphs produced by different experts, who have different ideas of what concepts are important and distinct and which concepts are important to understanding others; and (3) the concept graphs we learn from a collection of documents will differ significantly from those we imagine, without these differences necessarily being better or worse.

In this work, we assume that a topic model provides a reasonable proxy for the concepts a person might identify in a technical corpus. However, topic modeling approaches are better at finding general areas of research than at identifying fine-grained concepts like those shown in Figure 1. The concept graph formalism can be extended with the use of discrete entities, identified by a small set of names, e.g., (*First-order logic*, *FOL*). We have performed initial work on two approaches to extract entities:

1 We can use an external reference, Wikipedia, to help entity extraction. We count the occurrences of each article title in the scientific corpus, and we keep the high-frequency titles as entities. For example, in the ACL Anthology corpus, we obtain 56 thousand entities (page titles) that occurred at least once and 1,123 entities that occur at least 100 times.

2 We cannot assume that the important entities in every scientific or technical corpus will be well-represented on Wikipedia. In the absence of a suitable external reference source, we can use the open-source tool SKIMMR (Nováček and Burns, 2014) or the method proposed by Jardine (2014) to extract important noun phrases to use as entities. The importance of a potential entity can be computed based on the occurrence frequency and the sentence-level co-occurrence frequency with other phrases.

Another limitation of using a topic model like LDA as a proxy for concepts is that the topics are static, while a corpus may span decades of research. Studying how latent models might evolve or "drift" over time within a textual corpus describing a technical discipline is an important research question, and our approach could be extended to add or remove topics in a central model over time.

Despite its limitations, a topic model is useful for automatically discovering concepts in a corpus even if the concept is not explicitly mentioned in a document (e.g., the words "axiom" or "predi-

cate" might indicate discussion of logic) or has no canonical name. The concept graph representation allows for the introduction of additional or alternative features for concepts, making it suitable for new methods of identifying and linking concepts.

## 7 Conclusions

Problems such as reading list generation require a representation of the structure of the content of a scientific corpus. We have proposed the concept graph framework, which gives weighted links from documents to the concepts they discuss and links concepts to one another. The most important link in the graph is the concept dependency relation, which indicates that one concept helps a learner to understand another, e.g., *Markov logic networks* depends on *Probability*.

We have presented four approaches to predicting these relations. We propose information-theoretic measures based on cross entropy and on information flow. We also present baselines that compute the similarity of the word distributions associated with each concept, the likelihood of a citation connecting the concepts, and a hierarchical clustering approach. While word similarity proves a strong baseline, the strongest edges predicted by the cross-entropy approach are more precise. We are releasing human annotations of concept nodes and possible dependency edges learned from the ACL Anthology as well as implementations of the methods described in this paper to enable future research on modeling scientific corpora.[2]

## Acknowledgments

The authors thank Yigal Arens, Emily Sheng, and Jon May for their valuable feedback on this work.

This work was supported by the Intelligence Advanced Research Projects Activity (IARPA) via the Air Force Research Laboratory. 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, AFRL, or the U.S. Government.

[2]The code and data associated with this work are available at http://techknacq.isi.edu

## References

Sanjeev Arora, Rong Ge, and Ankur Moitra. 2012. Learning topic models – going beyond SVD. In *Proceedings of the 53rd Annual Symposium on Foundations of Computer Science*, pages 1–10. IEEE.

Steven Bird, Robert Dale, Bonnie Dorr, Bryan Gibson, Mark Joseph, Min-Yen Kan, Dongwon Lee, Brett Powley, Dragomir Radev, and Yee Fan Tan. 2008. The ACL Anthology Reference Corpus: A reference dataset for bibliographic research in computational linguistics. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation*, Marrakech, Morocco, May. European Language Resources Association.

David Blei and John Lafferty. 2006. Correlated topic models. In *Advances in Neural Information Processing Systems*.

David M. Blei, Andew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Eunsol Choi, Matic Horvat, Jon May, Kevin Knight, and Daniel Marcu. 2016. Extracting structured scholarly information from the machine translation literature. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*. European Language Resources Association.

Philipp Cimiano, Andreas Hotho, and Steffen Staab. 2005. Learning concept hierarchies from text corpora using formal concept analysis. *Journal of Artificial Intelligence Research*, 24(1):305–39, August.

Gregory Grefenstette. 1994. *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publishers, Norwell, MA, USA.

Qirong Ho, Jacob Eisenstein, and Eric P. Xing. 2012. Document hierarchies from text and links. In *Proceedings of the International World Wide Web Conference*, April.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 50–7. ACM.

James G. Jardine. 2014. Automatically generating reading lists. Technical Report UCAM-CL-TR-848, University of Cambridge Computer Laboratory, February.

Istvan Jonyer, Diane J. Cook, and Lawrence B. Holder. 2002. Graph-based hierarchical conceptual clustering. *Journal of Machine Learning Research*, 2:19–43, March.

Andrew McCallum. 2002. MALLET: A machine learning for language toolkit. http://mallet.cs.umass.edu.

Vít Nováček and Gully APC Burns. 2014. SKIMMR: Facilitating knowledge discovery in life sciences by machine-aided skim reading. *PeerJ*, 2:e483.

Dragomir R. Radev, Pradeep Muthukrishnan, Vahed Qazvinian, and Amjad Abu-Jbara. 2013. The ACL Anthology Network Corpus. *Language Resources and Evaluation*, pages 1–26.

Martin Rosvall and Carl T. Bergstrom. 2008. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4):1118–23.

Jean Michel Rouly, Huzefa Rangwala, and Aditya Johri. 2015. What are we teaching?: Automated evaluation of CS curricula content using topic modeling. In *Proceedings of the Eleventh Annual International Conference on International Computing Education Research*, pages 189–197.

Mark Sanderson and Bruce Croft. 1999. Deriving concept hierarchies from text. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 206–13, New York, NY, USA. ACM.

Ulrich Schäfer, Bernd Kiefer, Christian Spurk, Jörg Steffen, and Rui Wang. 2011. The ACL Anthology Searchbench. In *Proceedings of the ACL-HLT 2011 System Demonstrations*, pages 7–13.

Laura M. Smith, Linhong Zhu, Kristina Lerman, and Allon G. Percus. 2014. Partitioning networks with node attributes by compressing information flow. *arXiv preprint arXiv:1405.4332*.

Partha Pratim Talukdar and William W. Cohen. 2012. Crowdsourced comprehension: Predicting prerequisite structure in Wikipedia. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 307–15. Association for Computational Linguistics.

Xiaolong Wang, Chengxiang Zhai, and Dan Roth. 2013. Understanding evolution of research themes: A probabilistic generative model for citations. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1115–23, New York, NY, USA. ACM.

Ryen W. White and Joemon M. Jose. 2004. A study of topic similarity measures. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and development in Information Retrieval*, pages 520–1. ACM.

William A. Woods. 1997. Conceptual indexing: A better way to organize knowledge. Technical report, Sun Microsystems, Inc., Mountain View, CA, USA.

Yiming Yang, Hanxiao Liu, Jaime Carbonell, and Wanli Ma. 2015. Concept graph learning from educational data. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pages 159–68. ACM.