Beyond Captions: Linking Figures with Abstract Sentences in Biomedical Articles

Joseph P. Bockhorst[1]*, John M. Conroy[2], Shashank Agarwal[3], Dianne P. O'Leary[4], Hong Yu[1,3]*

[1] Department of Computer Science, University of Wisconsin–Milwaukee, Milwaukee, Wisconsin, United States of America, [2] IDA/Center for Computing Sciences, Bowie, Maryland, United States of America, [3] Department of Health Sciences, University of Wisconsin–Milwaukee, Milwaukee, Wisconsin, United States of America, [4] Computer Science Department and UMIACS, University of Maryland, College Park, Maryland, United States of America

Abstract

Although figures in scientific articles have high information content and concisely communicate many key research findings, they are currently under utilized by literature search and retrieval systems. Many systems ignore figures, and those that do not typically only consider caption text. This study describes and evaluates a fully automated approach for associating figures in the body of a biomedical article with sentences in its abstract. We use supervised methods to learn probabilistic language models, hidden Markov models, and conditional random fields for predicting associations between abstract sentences and figures. Three kinds of evidence are used: text in abstract sentences and figures, relative positions of sentences and figures, and the patterns of sentence/figure associations across an article. Each information source is shown to have predictive value, and models that use all kinds of evidence are more accurate than models that do not. Our most accurate method has an $F1$-score of 69% on a cross-validation experiment, is competitive with the accuracy of human experts, has significantly better predictive accuracy than state-of-the-art methods and enables users to access figures associated with an abstract sentence with an average of 1.82 fewer mouse clicks. A user evaluation shows that human users find our system beneficial. The system is available at http://FigureItOut.askHERMES.org.

Citation: Bockhorst JP, Conroy JM, Agarwal S, O'Leary DP, Yu H (2012) Beyond Captions: Linking Figures with Abstract Sentences in Biomedical Articles. PLoS ONE 7(7): e39618. doi:10.1371/journal.pone.0039618

Editor: Christos A. Ouzounis, The Centre for Research and Technology, Hellas, Greece

Received August 25, 2011; **Accepted** May 23, 2012; **Published** July 18, 2012

Funding: This study was supported by National Institutes of Health grants 5R01LM009836 and 5R21RR024933 and the Research Growth Initiative (RGI) from University of Wisconsin- Milwaukee. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: joebock@uwm.edu (JPB); hongyu@uwm.edu (HY)

Introduction

The rapid growth of electronic full-text biomedical articles has enabled the development of information systems that allow researchers to search and navigate large literature databases. Key content of many articles resides in images, charts, plots, tables or diagrams, and there is considerable interest in developing new figure aware systems. Because of the important role of figures, they often are referred to and discussed explicitly and implicitly throughout an article. However, nearly all existing systems for figure search rely solely on the text in captions, and thus fail to consider other key document elements. We present novel algorithms for automatically "linking" or "associating" sentences in the abstract of a scientific article with figures in the article body. These and related methods will help figures become a key part of next generation search systems. We use the terms "associating" and "linking" to indicate that a figure and a sentence in the abstract are related. In particular, the figure gives supporting information for the sentence in the abstract. This use of these terms is common in data mining and text analysis. It should not be confused with genetic, biological or medical uses of the terms "links" and "association".

Our approach uses three types of evidence to predict whether or not an abstract sentence is associated with a figure. The first type of evidence is text. While the textual representation of a sentence is simply the terms in the sentence, the appropriate textual representation of a figure is not so clear. We investigate textual figure representations based on terms in the figure's caption and/or its referencing paragraphs. We use probabilistic language models to assess the textual similarity between an abstract sentence and a figure. The second type of evidence is the relative positions of a sentence and figure. Previous work by our group [1] has shown that sentences at the beginning of an abstract are more likely to be associated with figures near the beginning of an article, middle sentences are more likely to be associated with middle figures, and so on. We use probabilistic distance models to reason about the relative positions for both linked and non-linked instances. The third type of evidence is patterns of sentence/figure links across an article. Since the presence or absence of a link for one instance can affect the likelihood of a link for other instances [1], we introduce novel approaches for representing linkage patterns based on hidden Markov models (HMMs) [2] and conditional random fields (CRFs) [3].

Our experimental evaluation uses a corpus of 114 biomedical articles annotated by their authors for all links between abstract sentences and figures. Figure 1 shows the annotated linkages between figures and the abstract sentences of one such article. We use supervised learning to learn language, distance, and linkage flow models, and use probabilistic methods to effectively combine predictions of the three models. Cross-validation experiments are used to evaluate our methods. The key findings are (i) each type of evidence has predictive value, (ii) predictions of models that combine evidence sources are more accurate than the predictions

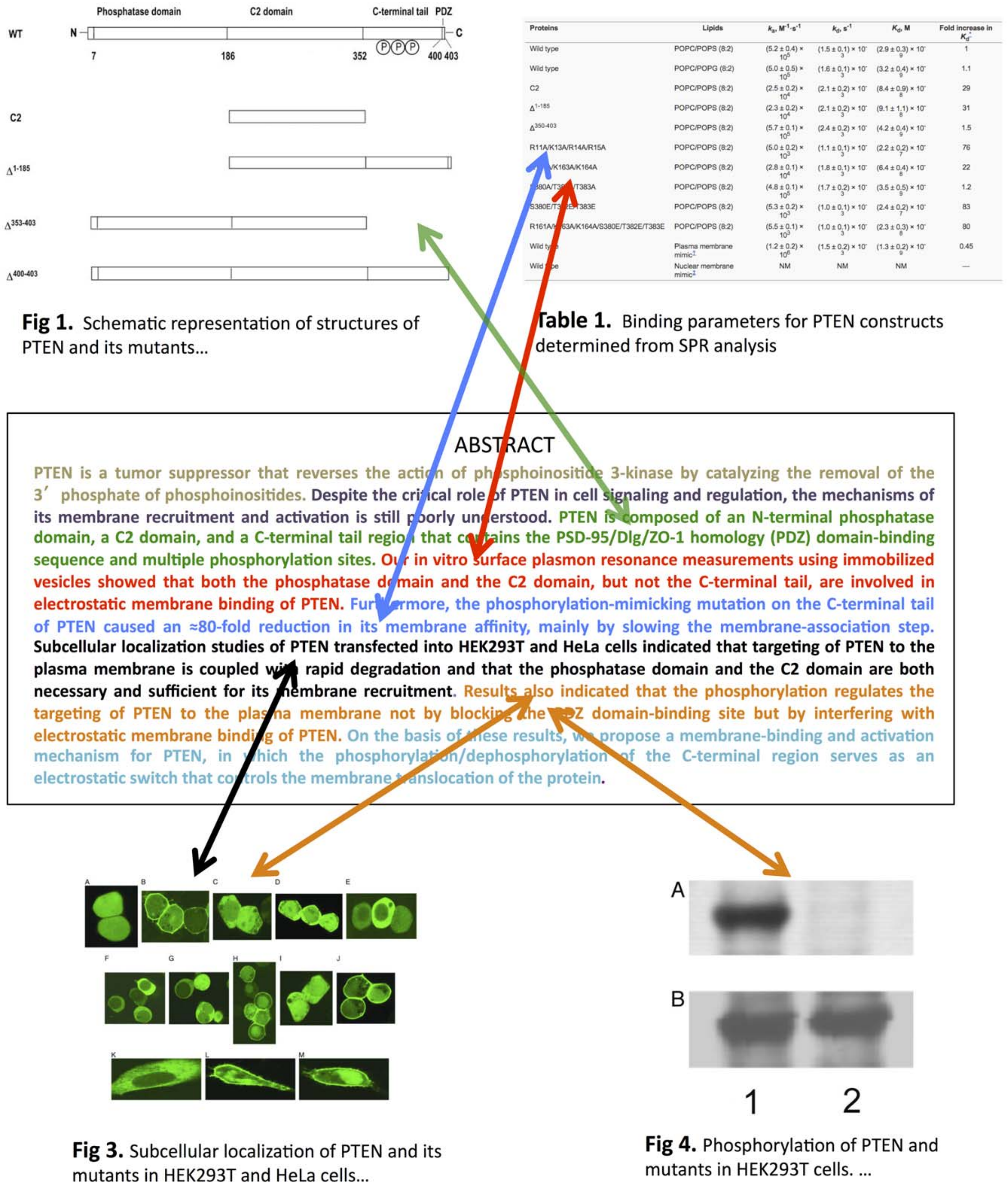

Fig 1. Schematic representation of structures of PTEN and its mutants...

Proteins	Lipids	k_a, $M^{-1} \cdot s^{-1}$	k_d, s^{-1}	K_d, M	Fold increase in K_d*
Wild type	POPC/POPS (8:2)	$(5.2 \pm 0.4) \times 10^5$	$(1.5 \pm 0.1) \times 10^{-3}$	$(2.9 \pm 0.3) \times 10^{-9}$	1
Wild type	POPC/POPG (8:2)	$(5.0 \pm 0.5) \times 10^5$	$(1.6 \pm 0.1) \times 10^{-3}$	$(3.2 \pm 0.4) \times 10^{-9}$	1.1
C2	POPC/POPS (8:2)	$(2.5 \pm 0.2) \times 10^4$	$(2.1 \pm 0.2) \times 10^{-3}$	$(8.4 \pm 0.9) \times 10^{-8}$	29
Δ^{1-185}	POPC/POPS (8:2)	$(2.3 \pm 0.2) \times 10^4$	$(2.1 \pm 0.2) \times 10^{-3}$	$(9.1 \pm 1.1) \times 10^{-8}$	31
$\Delta^{350-403}$	POPC/POPS (8:2)	$(5.7 \pm 0.1) \times 10^5$	$(2.4 \pm 0.2) \times 10^{-3}$	$(4.2 \pm 0.4) \times 10^{-9}$	1.5
R11A/K13A/R14A/R15A	POPC/POPS (8:2)	$(5.0 \pm 0.2) \times 10^3$	$(1.1 \pm 0.1) \times 10^{-3}$	$(2.2 \pm 0.2) \times 10^{-7}$	76
[illegible]/K163A/K164A	POPC/POPS (8:2)	$(2.8 \pm 0.1) \times 10^4$	$(1.8 \pm 0.1) \times 10^{-3}$	$(6.4 \pm 0.4) \times 10^{-8}$	22
[illegible]380A/T38[illegible]/T383A	POPC/POPS (8:2)	$(4.8 \pm 0.1) \times 10^5$	$(1.7 \pm 0.2) \times 10^{-3}$	$(3.5 \pm 0.5) \times 10^{-9}$	1.2
S380E/T3[illegible]E/T383E	POPC/POPS (8:2)	$(5.3 \pm 0.2) \times 10^3$	$(1.0 \pm 0.1) \times 10^{-3}$	$(2.4 \pm 0.2) \times 10^{-7}$	83
R161A/[illegible]63A/K164A/S380E/T382E/T383E	POPC/POPS (8:2)	$(5.5 \pm 0.1) \times 10^3$	$(1.0 \pm 0.1) \times 10^{-3}$	$(2.3 \pm 0.3) \times 10^{-8}$	80
Wild ty[illegible]	Plasma membrane mimic‡	$(1.2 \pm 0.2) \times 10^6$	$(1.5 \pm 0.2) \times 10^{-3}$	$(1.3 \pm 0.2) \times 10^{-9}$	0.45
Wild [illegible]pe	Nuclear membrane mimic‡	NM	NM	NM	—

Table 1. Binding parameters for PTEN constructs determined from SPR analysis

ABSTRACT

PTEN is a tumor suppressor that reverses the action of phosphoinositide 3-kinase by catalyzing the removal of the 3′ phosphate of phosphoinositides. Despite the critical role of PTEN in cell signaling and regulation, the mechanisms of its membrane recruitment and activation is still poorly understood. PTEN is composed of an N-terminal phosphatase domain, a C2 domain, and a C-terminal tail region that contains the PSD-95/Dlg/ZO-1 homology (PDZ) domain-binding sequence and multiple phosphorylation sites. Our in vitro surface plasmon resonance measurements using immobilized vesicles showed that both the phosphatase domain and the C2 domain, but not the C-terminal tail, are involved in electrostatic membrane binding of PTEN. Furthermore, the phosphorylation-mimicking mutation on the C-terminal tail of PTEN caused an ≈80-fold reduction in its membrane affinity, mainly by slowing the membrane-association step. Subcellular localization studies of PTEN transfected into HEK293T and HeLa cells indicated that targeting of PTEN to the plasma membrane is coupled with rapid degradation and that the phosphatase domain and the C2 domain are both necessary and sufficient for its membrane recruitment. Results also indicated that the phosphorylation regulates the targeting of PTEN to the plasma membrane not by blocking the PDZ domain-binding site but by interfering with electrostatic membrane binding of PTEN. On the basis of these results, we propose a membrane-binding and activation mechanism for PTEN, in which the phosphorylation/dephosphorylation of the C-terminal region serves as an electrostatic switch that controls the membrane translocation of the protein.

Fig 3. Subcellular localization of PTEN and its mutants in HEK293T and HeLa cells...

Fig 4. Phosphorylation of PTEN and mutants in HEK293T cells. ...

Figure 1. An example of a full-text biomedical article (pmid = 12808147) with author identified links between sentences in the abstract and figures and tables in the body of the article. Abstract sentences are shown in different colors. Arrows denote the annotated associations and arrow colors correspond to sentence color. To save space, figure captions are truncated and Fig. 2, which is not linked with any sentence, is not shown. (Figures republished with permission from [32], Copyright (2003) National Academy of Sciences, U.S.A.).
doi:10.1371/journal.pone.0039618.g001

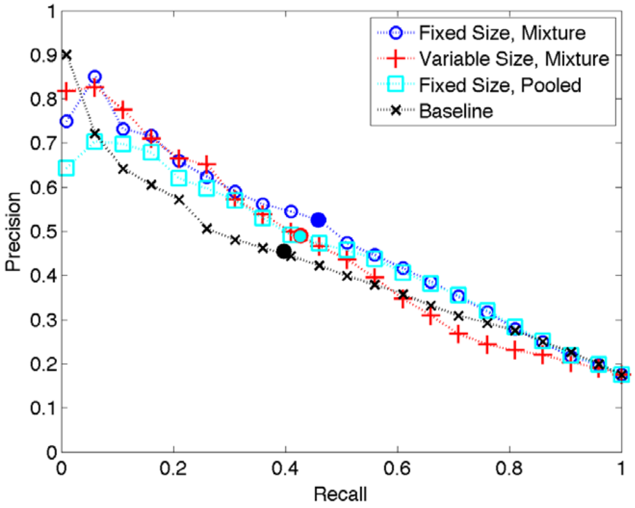

Figure 2. Recall-precision curves for three LMs and the baseline. The (Fixed Size, Mixture) model is our CompleteLM. The filled circles denote locations of the $Prec^J$ points.
doi:10.1371/journal.pone.0039618.g002

of models that use a single evidence source, (iii) across articles the average maximum F1 score of our combined approach is 69%, and (iv) our predictions would save users an average of 1.82 mouse clicks when searching for a figure associated with an abstract sentence in a conceptualized literature search system.

The work presented here extends previous work of our group on linking abstract sentences with figures [1] in several significant ways, and makes important contributions to our understanding of this problem. The present system uses supervised learning approaches while previous methods are unsupervised. We introduce probabilistic language models, position models, and HMM and CRF linkage flow models for this task. We evaluate two kinds of figure representations, one based on text in figure captions and the other on text in referencing paragraphs. A new evaluation measure based on the average number of saved clicks is introduced. Finally, the accuracy of predictions is significantly improved.

Our system is fully implemented and contains over 150,000 open access full-text biomedical articles that can be accessed at http://figureitout.askhermes.org.

Related Work

In this section we discuss relationships to previous work in four areas: text-based literature search systems, classification and search methods for images in documents, textual entailment, and summarization.

A number of medical and biological text-based literature search systems have been constructed. These include systems that respond to users' queries, such as PubMed and AskHERMES [4] for medical literature. Textpresso [5] was originally designed to assist biological database curation but also functions as an information retrieval system. Arrowsmith [6] helps biologists formulate hypotheses through text mining of two topics, such as a drug and an adverse event. Other systems attempt to find specific kinds of information. For example, GeneWays [7] extracts molecular interactions related to pathways identified in the literature and iHOP [8] identifies sentences that relate two genes. Additionally, there are numerous annotated databases – for example, the Gene Ontology annotation [9] the mouse Genome Database [10], SWISSPORT, OMIM [11], and BIND [12] – that provide different levels of annotated literature information about genes and molecular interactions. See the review article [13] for other information systems.

In addition to text, the importance of biomedical figures and images for document classification and retrieval has been recognized. The earliest image mining system is the Subcellular Location Image Finder (SLIF) system [14–18] which extracts and analyzes fluorescence microscope images from biomedical full-text articles. Other studies have looked at applying supervised machine-learning algorithms for image categorization using flat [19] and hierarchical [20] classification schemes. These methods showed that image classification benefits document classification. Besides the image content itself, associated text has been shown to be important for image mining. Caption words, for example, can improve image classification [19]. BioText searches biomedical images on the basis of image captions [21,22], and Yale image finder [23] searches images on the basis of title, abstract, image caption, and the text appearing in an image. More recently, an approach has investigated using figure-associated text for automatically ranking figures by their importance [24]. While these methods utilize text for different tasks, they do not automatically associate images or the figures that contain them with specific document text.

Our approaches of associating figures with text are also related to the problem of textual entailment [25], a task that has application to numerous higher-level problems including passage retrieval, machine translation, paraphrasing, summarization, and question answering. The PASCAL Network of Excellence Recognizing Textual Entailment (RTE) challenge task is to recognize whether two text strings can be semantically inferred (entailed) from each other. Thus, a body of text is said to "entail" a hypothesis text if the body of text implies that the hypothesis is true. Our task is similar in that the aim is to determine whether or not one string (a sentence in the abstract) is associated with another string (the text of a figure). The RTE task does not directly apply to the linkage between figures and text as the relationships between linked abstract sentences and figures is generally much weaker than entailment.

Lastly, we find similarities with the computational summarization work of Jing and McKeown [26]. They learn a summarization system from a training set consisting of human-written summary sentences in which words in the summaries are mapped to words in the original article. Their summarization approach, which assumes human summaries are created via a cut-and-paste process, uses two heuristic rules: (1) human summaries are more likely to use whole phrases than single, isolated words, and (2) humans are more likely to merge nearby sentences into a single sentence than combine sentences that are far apart. They model these tendencies with HMMs. These rules parallel patterns of associations between figures and abstract sentences that we represent with HMMs and CRFs. A key difference between these two tasks, however, is that while for summarization Jing and McKeown [26] permit only one-to-one associations between words in summary sentences and words in the original, we allow more general one to many, many to one, and many to many associations between figures and sentences. In this way then, our application represents a more challenging task.

Results and Discussion

We conducted experiments on a corpus of 114 manually annotated biomedical articles to empirically evaluate our approach to predicting linkages between abstract sentences and figures. Our experiments involve training models and making predictions from a progressively increasing number of evidence types. First, we consider only text, and evaluate predictions of our language models (LMs). Next, we add position evidence, and evaluate predictions of combined LMs and distance models (LM+DM). Last, we add (inferred) linkage evidence, and evaluate predictions of our hidden Markov model (HMM) and conditional random field (CRF) methods.

We designed our experiments to test several statistical hypotheses. Each experiment was evaluated using up to 3 performance metrics commonly used in information retrieval, as well as an application specfic performance value ("clicks"). For each test the null hypothesis is that two competing approaches have the same mean measure ($H_0 : \mu_1 = \mu_2$) and the alternative hypothesis is $H_a : \mu_1 \neq \mu_2$. We report $p-$values (the probability of the observed data under H_0) in all cases where $p < 0.1$) The three hypotheses were:

(1) The quality of predictions of our complete language model (CompleteLM) exceeds those of the state-of-the-art approach, which uses only text.

(2) Both linkage and positional features are predictive of abstract sentence/figure linkage and are complementary to text.

(3) The predictive performance of our HMM and CRF methods, which integrate text, linkage, and positional features exceeds the performance of the state-of-the-art approach.

Our empirical results support each of these hypotheses.

We measure performance using standard measures: precision is the fraction of linked figures that are correctly identified by a system, recall is the fraction of figures linked by a system that are truly linked figures, AROC is the area under the curve defining the false-positive rate as a function of recall, and F1 is the geometric mean of precision and recall.

We also use a new measure that we call "clicks". This is not actual clicks by a designated user but a mathematical model meant to estimate the savings over a user reading an abstract sentence and then selecting figures sequentially, looking for supporting information for the sentence. Our model assumes that figures are selected in order until the set of figures relevant to a given sentence is found. This may be an overestimate (if the user has visual clues or has already clicked on some of these figures for a previous sentence), or an underestimate (if the user clicks on all sentences, just to be sure, or clicks on the back button) but it does provide consistent criteria for evaluating methods. More precisely, for any sentence, we assume that if Figure k is the last figure (truly) linked to that sentence, a user without our system would click k times to retrieve the relevant figures, accessing the figures sequentially until obtaining the desired information. If our system scores all of the relevant figures for the sentence within its top q choices, then our user would click q times, and the number of clicks saved would be $k-q$. We define "clicks" to be this difference, averaged over all sentences in the set of abstracts. "Clicks" thus represents the average reduction in the number of mouse clicks needed by a user to locate a figure associated with an abstract sentence when the user clicks on figures in the order determined by linkage scores rather than sequentially.

Results for Language Models

We performed a leave-one-article-out cross-validation experiment to assess the performance of different LM approaches. We evaluated four types of figure-specific models (Caption Only, Referencing Only, Pooled, and Mixture) and two types of background models (Variable Size and Fixed Size) and compare to the current state-of-the-art [1] (Baseline). See the methods section for LM specifics. Since the Mixture figure-specific model considers more text than the other two methods and differentiates between caption and referencing text, and since the Fixed Size background model corrects for a bias that Variable Size has against long sentences, we sometimes refer to the LM (Fixed Size, Mixture) as the CompleteLM. We hypothesize CompleteLM will outperform the other LMs as well as Baseline.

These differing performances are given in Table 1 where the column headers on the per-article side of the the table have an over-bar and subscript a to indicate that the reported values are averages across articles. Broadly, note that the scores for per-article are uniformly higher for all methods and measures than their corresponding scores for the whole-corpus. So, indeed, the methods perform better on articles with fewer links. We will analyze these differences in detail using a permutation test, but first we discuss the results of using differing background models, i.e., differences between top and bottom rows of the table.

Table 1. Performance measures of text-only models.

Background	**Figure**	**whole-corpus**			**per-article**		
Model Vocab.	**Model Type**	$AROC$	$F1^*$	$Prec^J$	$\overline{AROC_a}$	$\overline{F1^*_a}$	$\overline{Prec^J_a}$
Variable Size	Caption Only	0.66	0.38	0.39	0.69	0.53	0.40
Variable Size	Referencing Only	0.67	0.38	0.36	0.73	0.56	0.44
Variable Size	Pooled	0.74	0.48	0.49	0.76	0.60	0.50
Variable Size	Mixture	0.73	0.47	0.49	0.76	0.61	0.50
Fixed Size	Caption Only	0.71	0.43	0.43	0.74	0.56	0.45
Fixed Size	Referencing Only	0.68	0.39	0.38	0.75	0.58	0.45
Fixed Size	Pooled	0.77	0.49	0.49	0.80	0.63	*0.53
Fixed Size	Mixture	**0.78**	**0.50**	**0.53**	**0.81**	***0.64**	****0.54**
Baseline		0.75	0.45	0.46	0.80	0.62	0.49

We show performance for our eight language models and the baseline. The first three result columns show overall corpus-wide performance, and the last three result columns show mean performance across articles. The first seven result rows show performance of incomplete LMs, and the eighth result row shows the performance of our CompleteLM, (Fixed Size, Mixture). Our CompleteLM performs best on all measures. Asterisks denote p-values from paired t-tests comparing each method with Baseline, where * indicates $p < 0.1$ and **indicates $p < 0.01$.
doi:10.1371/journal.pone.0039618.t001

Reported values in the table are the precisions that arise when the number of predicted linkages is equal to the number of abstract sentences. That is, the precision value $Prec^J(a)$ for article a (used in the calculation of $\overline{Prec_a^J}$ in the per-article method) is the precision for the top scoring $J(a)$ sentence-figure instances, where $J(a)$ is the number of abstract sentences in article a. Similarly, $Prec^J$ for the whole-corpus case is the precision for the top scoring $\sum_a J(a)$ instances. Figure 2 shows whole-corpus recall-precision curves for three LM models and the baseline.

We continue our discussion of results by comparing CompleteLM with Baseline. We observe that the performance of our approach exceeds the baseline on all measures. To estimate significance we conducted paired t-tests for the three per-article measures. The p-value for $\overline{F1_a^*}$ (0.063) nearly indicated significance at the standard 0.05 level, while the p-value for the important $\overline{Prec_a^J}$ case (0.0071) is significant. Thus, we conclude the expected value of $Prec_a^J$ for CompleteLM is larger than for Baseline. Since the $Prec_a^J$ measure, unlike $AROC_a$ and $F1_a^*$, depends only on labels of top scoring instances, the improvement in $Prec_a^J$ is especially relevant to literature browsing systems, which are likely to provide access to figures for only a few of the highest scoring instances. In the recall-precision curves (Figure 2) we observe that, except for very small recall levels less than 0.05, the CompleteLM curve dominates Baseline up to a recall of 0.8.

We now turn to a comparison of our eight LMs. We look first at the performance of different figure models. For a given background model, the LMs for pairings with Mixture and Pooled figure models have consistently better performance than pairings with both Caption Only and Referencing Only models. Paired t-tests on the per-article measures confirm that for all cases these differences are significant ($p \leq 0.01$). We conclude that our LM approach successfully combines complementary sources of text. Next, we compare our two background models. The Fixed Sized models, which have background vocabulary sizes set to a constant value because of a potential bias against linking long sentences, have better performance than Variable Size models when comparisons are made between pairings with the same figure model. Differences between Fixed Sized and Variable Size background models paired with Mixture figure models are significant (p-values of paired t-tests ≤ 0.01) for all three measures.

To investigate a possible bias favoring links to longer sentences we plot in Figure 3 empirical cumulative distribution curves of sentence lengths for four collections of sentence/figure instances: all 5402 instances (magenta line), all 947 linked instances (black line), and the top scoring 826 instances under (Variable Size, Mixture) (blue line) and (Fixed Size, Mixture) (red line) models. We choose 826 because this is the total number of abstract sentences in our corpus, and thus these curves show sentence length distribution at the $Prec^J$ point. The Variable Size method prefers linking short sentences rather than long sentences (gap between the red and black curves for a given sentence length). For example, although only 58% of linked instances have sentences with ten or fewer terms, 76% of all high-scoring instances under the Variable Size approach have sentences at least this short. The Fixed Size approach eliminates this bias, especially for sentences longer than 10 terms. Interestingly, there appears to be an actual preference *for* longer sentences to be linked as seen by comparing the magenta “All Instances” curve to the black “Linked Instances” curve. This may be reflective of a positive correlation between sentence length and information content.

Lastly, we look in more detail at uniformly higher performance in the per-article metrics by setting up an experiment to see if the differences are due to in-homogeneity of the data. That is, do some articles have language modelling scores that make linking sentences with figures easier? This may be due to an author's style or the topic being discussed. We explore this by employing a permutation test. We hold model type fixed and compare performance values calculated using the whole-corpus method with their per-article counterparts (that is, we compare values within rows of Table 1). We observe that for all three measures and all models the per-article performance value is larger than the whole-corpus value. While due to the way $\overline{F1_a^*}$ is calculated, we expect the larger per-article values for this measure, there is no calculation bias for area under the ROC curve and precision.

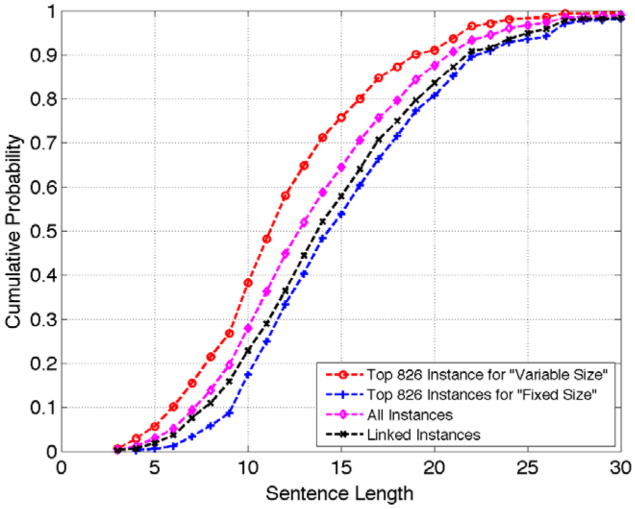

Figure 3. Empirical cumulative distribution functions of sentence length for four collections of instances: all 5406 instances, all 947 linked instances, and the top 826 scoring instances from (Fixed Size, Mixture) and (Variable Size, Mixture) language models.
doi:10.1371/journal.pone.0039618.g003

The permutation test shuffled the associations between articles and sentence/figure instances, keeping the number of linked instances associated with each article fixed. Although whole-corpus performance values do not depend on article assignments, per-article performance values do. For each of 1000 permutations we computed $\overline{AROC_a}$ and $\overline{Prec^J}_a$ from linkage scores of CompleteLM using the shuffled article associations. Figure 4 shows normalized histograms of observed performance values from the permutation test along with actual whole-corpus and per-article values. Although there is an article effect for both measures, it is clearest for AROC. On this measure, while the whole-corpus $AROC$ value of 0.777 was near the median (exceeding the per-article value in 485 of the 1000 permutations) the actual per-article $\overline{AROC_a}$ of 0.805 was substantially larger than any of the permuted values. On precision the whole-corpus $Prec^J$ value was larger than 904 of the permuted values, and the actual per-article $\overline{Prec^J}_a$ was larger than 999 of the 1000 permuted values. One consequence of the observed article effect is that since in a literature browsing system linkage scores are only considered one article at a time, whole-corpus performance measures will underestimate system performance in practice. A second and more important consequence is that a single, fixed threshold on linkage score separating positive and negative predictions is not appropriate. It will be too permissive for articles with score-increasing effects, and conversely too restrictive for articles with score-reducing effects.

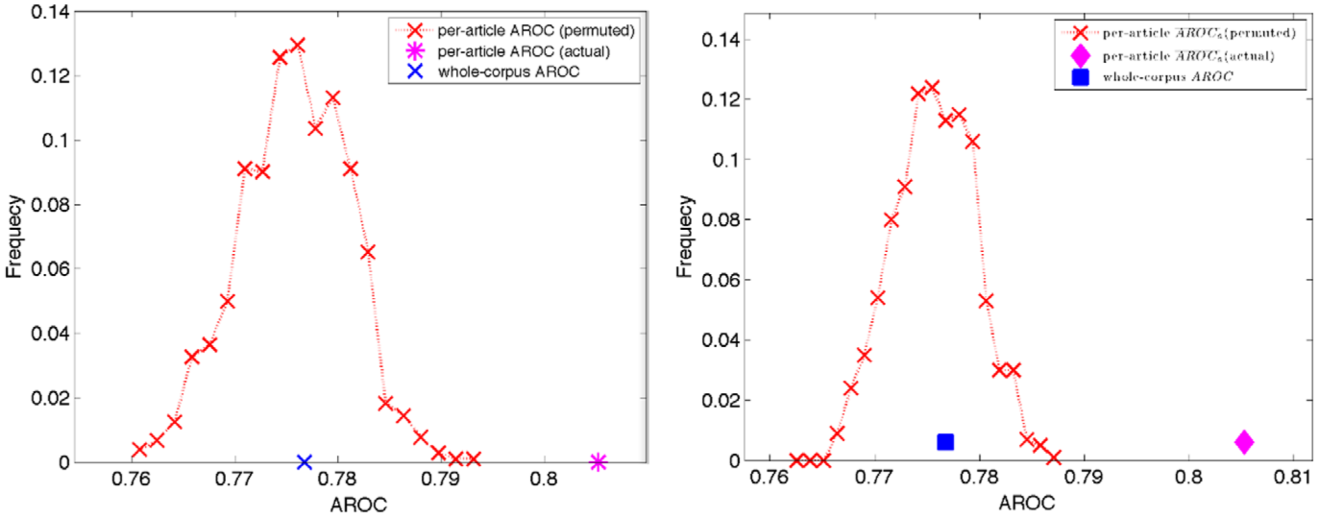

Figure 4. Results of permutation tests showing article-effects on two performance measures: area under the ROC curve (left) and precision (right). Blue and magenta points show actual performance values for the CompleteLM model calculated with the whole-corpus and per-article methods, respectively. The red-line shows a normalized histogram of per-article performance for 1000 random permutations of the associations between articles and abstract sentence/figure instances.
doi:10.1371/journal.pone.0039618.g004

Results for Combined Text and Non-text Models

In this section we evaluate approaches to linkage prediction that utilize both text and non-text features. We consider two kinds of non-text features. Values of *positional* features are based on the relative positions of the sentence within the abstract and the figure within the article, and values of *linkage* features are derived from linkage patterns of other instances in the same article. While the values of positional features are always observable, the values of linkage features are generally not observable when predictions are needed. Although the linkage values are hidden, since the hidden Markov model (HMM) and conditional random field (CRF) approaches collectively classify all of an article's instances simultaneously, inferred values of linkage features can inform their predictions. We first evaluate non-text features individually, and then in combination.

Information gain of non-text features. Table 2 shows the percent information gain (% gain) of non-text features. For comparison, we also show the % gain of the text-based CompleteLM model's linkage scores ($S^{LM}(j,k)$, see methods). A feature's % gain indicates how predictive of linkage that feature is in isolation. It ranges from 0% (not predictive) to 100% (completely predictive). It is not surprising that the % gain of CompleteLM scores (16.30%) is, by a wide margin, the largest, as these scores come from models of numerous text features while the other % gain values in the table are for individual features. After CompleteLM, the next three features form a well separated group with similar % gains. This group comprises two linkage features, EdgesCrossed (7.82%) and FigureDegree (7.37%), along with Distance (7.48%), a positional feature. The relatively high gain of Distance agrees with previous work [1] where we found that predictions based on text and Distance are more accurate than predictions of text-only models. The Distance feature is, however, the only non-text feature previously used for abstract sentence/figure linkage prediction. Therefore, the present work represents the first time the other features in Table 2 have been considered for this task.

Other than Distance, Initial Sentence, with a modest gain of 1.78%, is the only positional feature with % gain significantly different from 0.0. All linkage features, on the other hand, have statistically significant % gain values. Indeed, four linkage features have gains exceeding 4%. If appropriately modeled, these linkage features may lead to more accurate predictions of abstract sentence/figure associations. Incorporating them into a model, however, is challenging since linkage feature values are unobserved at prediction, and therefore approaches that predict linkages of each instance independently are unable to use linkage features. In fact, a key motivation behind our HMM and CRF approaches was to utilize their collective classification properties to model linkage features.

Evaluation of linkage predictions. To evaluate models of text and non-text features, we performed leave-one-article-out cross-validation experiments similar to those we used to evaluate language models. We look at three approaches to modeling text and non-text. Our CompleteLM+DM approach combines Com-

Table 2. Percent information gain of non-text features.

Feature Name	Feature Kind	% Gain
CompleteLM scores	–	*16.30
EdgesCrossed	linkage	*7.82
Distance	position	*7.48
FigureDegree	linkage	*7.37
PreviousFigure	linkage	*4.55
PreviousSentAndFig	linkage	*4.01
PreviousSentence	linkage	*1.78
InitialSentence	position	*1.57
SentenceDegree	linkage	*1.51
LastSentence	position	0.08
InitialFigure	position	0.00
LastFigure	position	0.00

For comparison with text features we also show the gain of CompleteLM scores. Asterisks indicate features whose % gain significantly differs from 0.0 (p-value of permutation test <0.01).
doi:10.1371/journal.pone.0039618.t002

pleteLM and distance model (DM) scores (Equation 13). Like the LM approaches, it predicts linkages independently for each sentence/figure pair. In contrast, our other two approaches, HMMs and CRFs, make collective predictions, and moreover utilize both positional and linkage non-text features. We evaluate both the sentences-in-states (SIS) and figures-in-states (FIS) HMM and CRF variants. We compare predictions of our models that merge text and non-text features to predictions of models that consider only distance (DM), only text (CompleteLM), and two baselines: the text-only Baseline described above and a combined text and distance method used in a previous study [1]. This text and distance baseline – called $SIM()$ by its authors, but which for consistency we refer to as Baseline+DM – represents the current state-of-the-art, and is currently the most accurate method for predicting abstract sentence/figure linkage. As above, we use the $AROC$, $F1^*$ and $Prec^J$ performance measures calculated corpus-wide and per-article. Additionally, we report per-article values of clicks, labeled $\overline{Clicks^J_a}$.

Table 3 shows the performance of various models, and Figure 5 shows whole-corpus recall-precision curves for a subset of models. CRF (SIS), our top performing model, has the highest performance on all measures. Paired t-tests indicate that differences between CRF (SIS) and Baseline+DM for all per-article measures are statistically significant (all p-values <0.001). From the recall-precision curves in Figure 5 we see that, except for recall levels <0.05, the CRF (SIS) curve dominates the Baseline+DM curve. Therefore, we conclude that CRF (SIS) represents a significant improvement over the state-of-the-art for predicting linkages between abstract sentences and figures.

Comparing the CRF and HMM methods, we observe that CRFs usually, but not always, outperform HMMs for the same variant (either SIS or FIS). The HMM (SIS) has higher precision than CRF (SIS) for recall levels below about 0.4 while CRF (SIS) has the higher precision at higher recall levels. However, when we compare the SIS and FIS constructions we see that HMM (SIS), the least performing SIS construction, outperforms both FIS approaches. Hence, it appears model variant (SIS or FIS) has more effect on performance than model type (HMM or CRF). Even so, the differences between CRF (SIS) and HMM (SIS) are significant (p-value $<$ 0.05) for $AROC_a$ and LM^*_a (but not $Prec^J_a$). Since aspects of the FigureDegree feature are captured by the SIS CRF but not the FIS variant, the superiority of the performance of SIS over FIS agrees with the relative information gain values (Table 2) of FigureDegree (7.37%) and SentenceDegree (1.51%). To better understand the performance differences between CRF (SIS) and CRF (FIS), we compared articles for which CRF (SIS) had larger $Prec^J_a$ score to those where CRF (FIS) had the higher score. Articles won by CRF (FIS) had on average of 1.11 fewer abstract sentences than articles won by CRF (SIS). A permutation test reveals that this 1.11 sentence difference is statistically significant. This suggests a suite of models approach, where the model applied to linkage prediction on a given article depends on the number of abstract sentences or other observable article properties, may be effective.

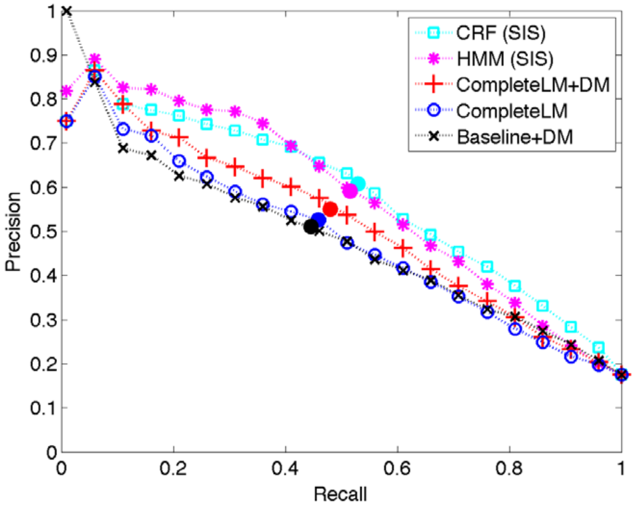

Figure 5. Whole-corpus recall-precision curves. The solid dots indicate the recall-precision point at $Prec^J$, when the number of predicted linked instances is equal to the total number of abstract sentences in the corpus.
doi:10.1371/journal.pone.0039618.g005

The overall trend evident in the performance measures of Table 3 and the recall-precision curves in Figure 5 is that performance increases as more types of features are utilized. Of the models that use a single class of features, those that use text only are clearly superior to the DM approach. Combining DM with text models gives a substantial performance boost, most markedly for Baseline+DM versus Baseline. We see another performance bump for models that incorporate linkage features. Thus, we conclude that text, positional and linkage features are complementary for linkage prediction, and that our approaches successfully integrate these diverse types of evidence.

Results for Human Annotators

We invited authors of a disjoint set of 49 additional PNAS articles to provide annotations of abstract sentence/figure associations for their articles, and to evaluate a prototype of our online article browsing system on their articles. We subsequently asked authors to complete a short four question usability survey. A total of 21 authors participated for a response rate of 43%. Further, we asked three bio-medical researchers who are not authors of any of these articles to provide additional annotations from which we obtained linkage annotations for 14 of these articles.

The 14 articles annotated by both authors and non-authors contain a total of 420 abstract sentence/figure instances. Table 4(a) shows the contingency table for the linkage annotations of these instances. Authors and non-authors have a related concept of sentence/figure association ($p<0.001$ for χ^2 test on independence of counts in Table 4). Authors and non-authors agree on linkage status on 81% of instances, and inter-annotator agreement as measured by Cohen's κ is 0.47. The concept of association, however, is not precise as non-authors and authors disagree 19% of the time. It is interesting that non-authors, with a 27% linked rate, appear to have a significantly more liberal notion of association than authors, who identify only 17% of instances as being linked.

Besides estimating inter-annotator agreement we can use non-author annotations to compare the computational predictions of our models with human predictions. Using author annotations as ground truth, we compare the performance of linkage predictions made by humans (*i.e.*, non-authors) with computational predictions. Using the CRF (SIS) model trained from author annotations for the 114 articles used above, we predicted linkages for the 14 articles that have both author and non-author annotations. For an article with J abstract sentences we predict that the top scoring J instances are linked and that the other instances are not linked. Thus, the CRF (SIS) row of Table 4 are $\overline{Recall^J_a}$, $\overline{Prec^J_a}$ and $\overline{F1^J_a}$ values. We point out that there is a difference between the $\overline{F1^J_a}$

Table 3. Performance values for models that use combinations of text, positional and linkage features.

				whole-corpus			per-article			
Method	**T**	**P**	**L**	$AROC$	$F1^*$	$Prec^J$	$\overline{AROC_a}$	$\overline{F1^*_a}$	$\overline{Prec^J_a}$	$\overline{Clicks^J_a}$
DM		✓		0.67	0.39	0.26	0.70	0.46	0.31	0.97
CompleteLM	✓			0.78	0.50	0.53	0.81	0.64	0.54	1.52
CompleteLM+DM	✓	✓		0.80	0.53	0.55	0.82	0.67	0.58	1.65
HMM (FIS)	✓	✓	✓	0.80	0.54	0.57	0.81	0.66	0.57	1.74
CRF (FIS)	✓	✓	✓	0.81	0.54	0.56	0.83	0.67	0.58	1.75
HMM (SIS)	✓	✓	✓	0.82	0.56	0.59	0.84	*0.68	*0.60	1.57
CRF (SIS)	✓	✓	✓	**0.84**	**0.57**	**0.61**	**0.86	**0.69	**0.62	**1.82
Baseline	✓			0.75	0.45	0.46	0.80	0.62	0.49	1.49
Baseline+DM	✓	✓		0.79	0.49	0.51	0.83	0.65	0.57	1.65

Values calculated across the whole-corpus as well as per-article averages are shown. Each row gives the performance measures for one model. The **T**, **P** and **L** columns indicate with a $\surd$ those models that use text, positional, and linkage features, respectively. The sentences-in-states CRF approach, CRF (SIS), has the top performance on all measures. Asterisks denote p-values (*denotes $p<0.1$ and **denotes $p<0.01$) from paired t-tests comparing the per-article performance of our methods to those of Baseline+DM, the current state-of-the-art.

doi:10.1371/journal.pone.0039618.t003

measure in Table 4 (average of $F1$ calculated at one predetermined point in each article) and the $\overline{F1}^*_a$ measure in Tables 1 and 3 (average of maximum $F1$ value in each article). While human annotators have higher performance values on all measures (none of these differences are statistically significant), the performance of CRF (SIS) is competitive with that of the non-author humans. On 9 of the 14 articles the human had the higher $F1$ score, while on the other five articles CRF (SIS) had higher $F1$.

Table 5 shows the pilot survey questions and average response values. We observe that users tend to have a positive view of the accuracy and usability of the prototype system. Interestingly, there is a significant positive correlation between an author's score for Q2 ("How useful are current figure-sentence associations?") and the $F1$ score of the system predictions for their article ($\rho=0.44, p=0.04$). Similar correlations were observed for other questions. From these results we conclude that methods for making more accurate predictions of sentence/figure associations, including the computational approaches we describe in this article, will lead to more usable online literature browsing systems.

Table 4. Results for 14 articles with human annotations provided by both authors and non-authors, and computational predictions provided by the CRF (SIS) model.

(a)		**Authors**		
		Non-linked	**linked**	**Total**
Non-Authors	Non-linked	283	22	305
	Linked	58	57	115
	Total	341	79	420
(b)	$\overline{Recall_a}$	$\overline{Prec_a}$	$\overline{F1_a}$	
Human	0.65	0.48	0.53	
CRF (SIS)	0.58	0.43	0.47	

(a) Contingency table of human annotations.
(b) Per-article average recall, precision and F1 score of non-author human annotations and computational predictions using author annotations as ground truth.

doi:10.1371/journal.pone.0039618.t004

Conclusion. We have described methods for computationally identifying associations between sentences in the abstract of a scientific article and figures (and tables) in the article body. We use supervised methods for learning. Our models use three types of evidence to predict whether or not an abstract sentence is linked with a figure: text (in the abstract sentence, figure caption, and passages that refer to the figure), the relative positions of the abstract sentence and figure, and patterns of inferred associations for other sentence/figure pairs in the article.

Each type of evidence has predictive value. Our experimental evaluation showed that models that use all evidence types are more accurate than models that use only one or two types of evidence. Our best performing models, based on conditional random fields (CRFs) [3], achieve a macro-average F1 score of 0.69. The area under its ROC curve is 0.86. These performance measures represent a statistically significant improvement on the state-of-the-art for this task, an unsupervised approach developed earlier [1]. Moreover, disagreement of human annotators on linkage status is nearly as common as prediction errors of our system.

We observed that the use of a language model significantly improved the results of previous work, where a TFIDF cosine similarity was used. Once a larger data set is collected and more detailed user feedback is assembled, a natural area of future exploration is more sophisticated language models. For example, the use of word bigram models, smoothing based on related clusters of articles, and divergence metrics such as Jensen-Shannon are all possible extensions of this work [27].

Automatic methods for predicting linkages between abstract sentences and figures are important for the development of the next generation of literature search and browsing systems. A user

Table 5. Survey questions and average response values.

Q1:	How accurate are the figure-sentence associations? (3.76)
Q2:	How useful are the current figure-sentence associations? (3.62)
Q3:	If the system is implemented, how eager will you be to use it? (3.57)
Q4:	Do you like the interface design? (4.10)

Values range from 1 (not at all) to 5 (very).

doi:10.1371/journal.pone.0039618.t005

study showed that users find the figure browsing features supported by our linkage predictions to be helpful. We have incorporated linkage predictions into our system (http://FigureItOut.askHERMES.org).

Methods

Data and Features

We re-use our collection of 114 full-text biomedical articles (39 from Cell, 29 from EMBO, 30 from the Journal of Biological Chemistry, and 16 PNAS) from our previous study [1]. The authors manually annotated their articles by identifying associations between abstract sentences and figures. The collection has 826 abstract sentences, 741 figures, and 5402 total sentence/figure instances of which 947 (17.5%) are linked. Of the abstract sentences 271 (32.8%) are not linked with any figure, 317 (38.4%) are linked with a single figure, and 238 (28.8%) are linked with multiple figures. And for figures, 91 (12.3%) are not linked with any abstract sentence, 423 (57.1%) are linked with a single sentence, and 227 (30.6%) are linked with multiple sentences. The range of the number of abstract sentences and figures in an article is [3,13] and [3,11], respectively.

Term vectors. We represent the text content of captions and referencing paragraphs with the "bag-of-words" representation, and for abstract sentences we use the "set of words" representation. The term vector $\vec{T}$ for a sentence or figure has V elements, one element for each term in the vocabulary. For figures, $T(t)$ is the number of occurrences of term t, while for sentences it is a binary indicator of the presence (1) or absence (0) of t.

Positional features. In addition to text we also use non-text features. The features naturally divide into two groups, positional features and linkage features. The value of the positional features for sentence j and figure k in article a depends on the positions j, k and the total number of abstract sentences (J) and figures (K) in a. We number sentences and figures sequentially as they appear in the article. So, for example, instance (j,k) is for the j^{th} abstract sentence and k^{th} figure in the article.

- $Distance(j,k) = |\frac{j}{J} - \frac{k}{K}|$. This feature measures the difference of the relative sentence and figure positions. It is the only non-text feature previously used for predicting sentence/figure linkages.
- $\mathrm{InitialSentence}(j) = 1$ if $j = 1$ and 0 otherwise.
- $\mathrm{LastSentence}(j) = 1$ if $j = J$ and 0 otherwise.
- $\mathrm{InitialFigure}(k) = 1$ if $k = 1$ and 0 otherwise.
- $\mathrm{LastFigure}(k) = 1$ if $k = K$ and 0 otherwise.

Linkage features. We compute the value of linkage features from article-wide linkage patterns. We represent the linkage of an article with the J-by-K linkage matrix $\mathcal{L}$, where $\mathcal{L}(j,k) = 1$ if sentence j is linked with figure k, and 0 otherwise. Figure 6 shows an example, which we use in the following six definitions of linkage features.

- $\mathrm{PreviousSentence}(j,k) = \mathcal{L}(j-1,k)$ (undefined when $j = 1$). This feature indicates if k links with the abstract sentence previous to j. The value of $\mathrm{PreviousSentence}(2,3)$ is 0 because sentence 1 and figure 3 are not linked.
- $\mathrm{PreviousFigure}(j,k) = \mathcal{L}(j,k-1)$ (undefined when $k = 1$). This feature indicates if j links with the figure previous to k. The value of $\mathrm{PreviousFigure}(2,3)$ is 0 because sentence 2 and figure 2 are not linked.
- $\mathrm{PreviousSentAndFig} = \mathcal{L}(j-1,k-1)$ (undefined if $j = 1$ or $k = 1$). This feature indicates if the previous sentence and figure are linked. The value of $\mathrm{PreviousSentAndFig}(2,3)$ is 1 because sentence 1 and figure 2 are linked.
- $\mathrm{FigureDegree}(j,k) = \sum_{j' \neq j} \mathcal{L}(j',k)$. This feature is the number of sentences (other than j) linked with figure k. The value of $\mathrm{FigureDegree}(2,3)$ is 1 because sentence 4 and figure 3 are linked.
- $\mathrm{SentenceDegree}(j,k) = \sum_{k' \neq k} \mathcal{L}(j,k')$. This feature is the number of figures (other than k) linked with sentence j. The value of $\mathrm{SentenceDegree}(2,3)$ is 0 because sentence 2 does not link with any other figure.
- Edges Crossed(j, k) =

$$\sum_{j'=1}^{j-1} \sum_{k'=k+1}^{K} \mathcal{L}(j',k') + \sum_{j'=j+1}^{J} \sum_{k'=1}^{k-1} \mathcal{L}(j',k').$$

This feature is the number of links inconsistent with the preservation of relative ordering across links. The name EdgesCrossed comes from number of edges that would be crossed by the edge $j---k$ in the graph representation of $\mathcal{L}$. In the example in Figure 6, the value of $\mathrm{EdgesCrossed}(2,3)$ is 1 because the edge $2---3$ crosses the single edge $4---1$, and $\mathrm{EdgesCrossed}(3,1)$ is 2 because the edge $3---1$ would cross 2 edges.

Since $\mathcal{L}$ is not observed while predicting, linkage feature values are also hidden. Therefore, these features are not helpful in methods that predict instance linkages independently. The inferred values of linkage features may, however, benefit prediction by techniques like our HMM and CRF approaches.

Language Models

We model text properties of linked and non-linked instances using probabilistic language models (LM). Our LM approach is motivated by the successful application of similar methods to document retrieval [27–29]. For document retrieval the LM approach induces for each document a probability model over all terms in the vocabulary. Then, a document's relevance to a query is defined as the probability of the query under its model.

Hiemstra's LM approach [28] uses two kinds of term distributions: a single background distribution $\tilde{b}$ shared by all documents, and a set of document-specific distributions

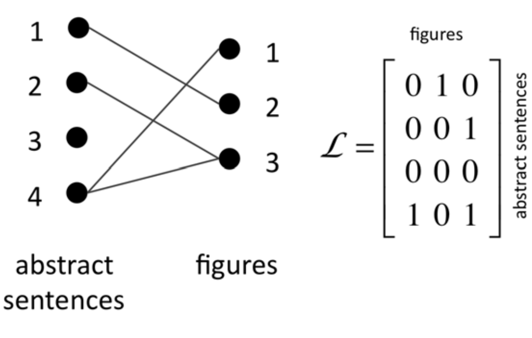

Figure 6. Example graph and linkage matrix representations for an article with four abstract sentences, three figures and four sentence/figure links. Combinations of linkages that induce edges that cross in the graph representation, {(4–1),(1–2)} and {(4–1),(2–3)} in this example, are less common as they are out of keeping with the observed tendency for consistent relative ordering among linked instances.

doi:10.1371/journal.pone.0039618.g006

$\tilde{d}_1, \tilde{d}_2, \cdots \tilde{d}_D$ for a D-document corpus. The LM represents the probability of query terms for document z as a mixture of $\tilde{b}$ and $\tilde{d}_z$. This mixture distribution corresponds to a generative process for constructing query terms for z that first randomly selects either $\tilde{b}$ or $\tilde{d}_z$ according to the mixing distribution (parameterized by λ) and then samples a term from the chosen distribution. To generate a query with L terms, these two steps are repeated L times.

In a similar way we use language models to predict links between abstract sentences and figures by treating abstract sentences as queries and figures as documents. Let $\tilde{b}(t)$ and $\tilde{d}_k(t)$ be the probability of term t under the background distribution and figure k's distribution respectively. Then, the probability of abstract sentence j given that it is linked to figure k is.

$$\Pr(\vec{T}_j | j \leftrightarrow k) = \prod_{t=1}^{V} \left[\lambda \tilde{b}(t) + (1-\lambda)\tilde{d}_k(t)\right]^{T_j(t)}, \tag{1}$$

where $\vec{T}_j$ is sentence j's length V term vector, $0 \le \lambda \le 1$ is the mixing proportion for the background distribution, and $j \leftrightarrow k$ denotes that j and k are linked, or equivalently that $\mathcal{L}(j,k) = 1$. If j and k are not linked, the background distribution generates all terms in the sentence:

$$\Pr(\vec{T}_j | j \not\leftrightarrow k) = \prod_{t=1}^{V} \left[\tilde{b}(t)\right]^{T_j(t)}. \tag{2}$$

The LM score matrix S^{LM} for an article holds the log-odds of the sentence terms given linkage for all instances (j,k). For an article with J sentences and K figures, S^{LM} is J-by-K and.

$$\mathrm{S}^{LM}(j, k) - \ln\left(\frac{\Pr(\overrightarrow{T}_j | j \leftrightarrow k)}{\Pr(\overrightarrow{T}_j | jk)}\right). \tag{3}$$

Figure-specific models. A natural and often-used representation of the document-specific term distribution $\tilde{d}$ is a multinomial distribution where each probability $\tilde{d}(t)$ has its own parameter. Parameter estimation for the multinomial model typically treats all occurrences of t in the document equally, and sets $\tilde{d}(t)$ to its frequency in the document. We consider multinomial representations, but we also use a representation that distinguishes caption terms from terms in referencing paragraphs.

Since in our approach to the linkage prediction task, figures play a role analogous to documents, to apply the multinomial approach we need to determine which terms represent a figure. Candidate term sources include terms in the figure's caption as well as terms in the article body close to figure references. We consider three sources: caption terms (Caption Only), terms in referencing paragraphs (Referencing Only) and the combination of terms in either the caption or a referencing paragraph (Pooled).

Let $n_k^c(t)$ be the number of occurrences of term t in figure k's caption and N_k^c be the total number of terms in the caption. We similarly define $n_k^r(t)$ and N_k^r for figure k's referencing paragraphs. The probabilities $d_k^c(t), d_k^r(t)$ and $d_k^p(t)$ of term t in the Caption Only, Referencing Only, and Pooled representations are simply its frequency in each collection:

$$d_k(t) = d_k^c(t) \leftarrow \frac{n_k^c(t)}{N_k^c} \quad \text{CaptionOnly} \tag{4}$$

$$d_k(t) = d_k^r(t) \leftarrow \frac{n_k^r(t)}{N_k^r} \quad \text{ReferencingOnly} \tag{5}$$

$$d_k(t) = d_k^p(t) \leftarrow \frac{n_k^c(t) + n_k^r(t)}{N_k^c + N_k^r}. \quad \text{Pooled} \tag{6}$$

Note that we do not use pseudo-counts here, as smoothing is unnecessary because the background distribution $\tilde{b}$ is used for terms that have zero probability in the figure-specific model.

Although the Pooled method has the advantage of including text from multiple sources, it is limited in that it ignores term origin even though there may be meaningful differences between terms in captions and referencing paragraphs. For example, while text in referencing paragraphs can discuss topics unrelated to the figure, caption content nearly always relates to the figure. Our final figure-specific term distribution, which we call Mixture, distinguishes between caption terms and referencing paragraph terms. In the Mixture approach we represent $d_k(t)$ itself as a mixture of the Caption Only and Referencing Only distributions,

$$d_k(t) = d_k^m(t) = \alpha d_k^c(t) + (1-\alpha) d_k^r(t), \quad \text{Mixture} \tag{7}$$

where $0 \le \alpha \le 1$ is the mixing proportion for the caption distribution.

Background models. We consider two approaches to setting the background distribution of article a. One approach pools all terms present in abstract sentences, figure captions and referencing paragraphs in a, and sets background probabilities to the smoothed term frequencies,

$$\tilde{b}(t) = \tilde{b}^v(t) \leftarrow \frac{n(t) + 1}{\left[\sum_{t'=1}^{V_a} n(t')\right] + V_a}, \quad \text{VariableSize} \tag{8}$$

where $n(t)$ is the count of term t in the pooled collection and V_a is the number of distinct terms in article a's sentence/caption/referencing paragraph pool. Since the vocabulary size V_a - which depends on the number of distinct terms in abstract sentences, captions and referencing paragraphs - varies from article to article, we call this approach to setting the background distribution VariableSize. Because of finite sampling, however, Equation 8 may lead to biases that favor linking short sentences and against linking long sentences. This bias arises because the probabilities $\tilde{b}(t)$ of terms present in the pooled collection set according to Equation 8 are too large. The $\tilde{b}(t)$ tend to be overestimates because, since the pooled collection is unlikely to contain all terms in the vocabulary, any term not in the pool has (an implicit) background probability of zero. Therefore, the probabilities of the absent terms are underestimated, and their true probability mass is distributed among the probabilities of the present terms. From Equations 1-3 it can be seen that overestimates of $\tilde{b}(t)$ cause a corresponding overestimate of $\Pr(\vec{T}_j | j \not\leftrightarrow k)$ (and thus underestimation of $\mathbf{S}^{LM}$) that increases with sentence length.

To correct for the bias in VariableSize, we consider an alternative approach to estimating $\tilde{b}$ that uses a fixed vocabulary size of Z terms in all articles. We call this approach FixedSize. We use a pseudo-count of 1 for all terms, and set the background probability for term t to.

$$\tilde{b}(t)=\tilde{b}^f(t)\leftarrow \frac{n(t)+1}{\left[\sum_{t'=1}^{V} n(t')\right]+Z}. \quad \text{FixedSize} \tag{9}$$

We describe below how we set Z from training sets.

Learning language models. We evaluate our LMs with leave-one-article-out cross-validation experiments. Our experiments evaluate each of the eight kinds of LMs: one LM for each pairing of a figure-specific-model (four kinds) with a background model (two kinds). For the cross-validation fold in which article a is in the test set, since the background and figure-specific distributions for a are set from only the terms in a and not any linkages, the parameters in $\tilde{b}$ and the $\tilde{d}_k$ do not depend on the training set of annotated articles. We do, however, use training sets to estimate our other LM parameters: the mixing proportions λ and α, and the fixed-vocabulary size Z.

We estimate separate parameter values for each LM. We first set λ's for the (VariableSize, Caption Only), (VariableSize, Referencing Only) and (VariableSize, Pooled) approaches. We search over 99 values of λ equally spaced from 0.01 to 0.99, and set λ to the value that maximizes the mean $\overline{Prec_a^J}$ on the training set. Next, we set Z for the (FixedSize, Caption Only), (FixedSize, Referencing Only) and (FixedSize, Pooled) models. For each figure-specific-model we temporarily set λ equal to the value just set for its pairing with VariableSize, and then estimate Z by the value that minimizes the absolute value of the Pearson correlation between sentence length and $\Pr(j\leftrightarrow k|\vec{T}_j)$ for all training instances (j,k). With Z set, we then estimate λ for these three LMs as we do above, by the value that minimizes the mean $\overline{Prec_a^J}$. Lastly, we set parameters of LMs with Pooled figure-specific-models. We set λ and α for (VariableSize, Pooled) with a method similar to the method we use to set λ for the other VariableSize models, though now we compute $\overline{Prec_a^J}$ for joint λ, α settings. So as to maximize the diversity of our parameter search, we define $\lambda_1=\lambda$, $\lambda_2=\lambda_1\alpha$, $\lambda_1+\lambda_2+\lambda_3=1$, and conduct our search on 120 evenly spaced points on the standard 2-simplex: $\{\lambda_1,\lambda_2,\lambda_3|\lambda_1\geq 0,\ \lambda_2\geq 0,\ \lambda_1+\lambda_2\leq 1\}$. Finally, we set Z, λ and α for (FixedSize, Pooled) analogous to the method we use above to set Z and λ for VariableSize models. We first set Z by minimizing correlation between sentence length and score, and next set λ and α by search on the 2-simplex.

Distance Model

We begin our description of non-text models with models of the Distance feature. We consider distance models (DMs) because it has been shown previously [1] that the relative positions of an abstract sentence and figure correlate with linkage status. For example, a sentence near the beginning of an abstract is more likely to be linked with a figure near the beginning of an article than with a figure at the end of the article.

We learn models of discretized values of the $\text{Distance}(j,k)$ feature for linked and non-linked instances. Let $\text{Distance}^{\bullet}(j,k)$ denote the bin of $\text{Distance}(j,k)$ where we have ten bins, and we place bin boundaries so that an approximately equal number of points falls in each bin. We set the bin probability of bin i in the DM of linked instances to the Laplace smoothed fraction of linked instances with $\text{Distance}^{\bullet}(j,k)=i$,

$$\Pr(\text{Distance}^{\bullet}(j,k)=i|j\leftrightarrow k)=\frac{n^L(i)+1}{\sum_{i'=1}^{10}\left(n^L(i')+1\right)}, \tag{10}$$

where $n^L(i)$ is the number of linked training set instances in bin i. We set bin probabilities for the DM of non-linked instances in a similar way,

$$\Pr(\text{Distance}^{\bullet}(j,k)=i|j\not\leftrightarrow k)=\frac{n^N(i)+1}{\sum_{i'=1}^{10}\left(n^N(i')+1\right)}, \tag{11}$$

where $n(i)^N$ is the number of non-linked training set instances in bin i.

The distance model score matrix $\mathbf{S}^D$ for an article holds the DM log-odds of the article's sentence/figure instances,

$$S^D(j,k)=\ln\left(\frac{\Pr(\text{Distance}^{\bullet}(j,k)|j\leftrightarrow k)}{\Pr(\text{Distance}^{\bullet}(j,k)|j\not\leftrightarrow k)}\right). \tag{12}$$

We construct scores of a combined language and distance model by adding scores,

$$\mathrm{S}^{LM+D}(j,k)=\ln\left(\frac{\Pr\left(\vec{T}_j,\text{Distance}^{\bullet}(j,k)|j\leftrightarrow k\right)}{\Pr\left(\vec{T}_j,\text{Distance}^{\bullet}(j,k)|j\not\leftrightarrow k\right)}\right), \tag{13}$$

$$=\mathbf{S}^{LM}(j,k)+\mathbf{S}^D(j,k) \tag{14}$$

where $\mathbf{S}^{LM+D}(j,k)$ is the combined LM and DM score for sentence j and figure k, and Equation 14 follows from Equations 3 and 12 under the assumption that terms and distances are conditionally independent given linkage status.

Hidden Markov Models

In addition to patterns of the Distance feature for individual instances, linkage patterns across instances also have tendencies. For example, given two linked instances, $j\leftrightarrow k$ and $j'\leftrightarrow k'$, from the same article, if $j>j'$, then it is also likely that $k>k'$. We model these kinds of linkage-flow patterns flow using hidden Markov models (HMMs) [2] and Conditional random fields (CRFs) [3], two kinds of probabilistic models widely used for representing structure in sequential problems. Since flow tendencies indicate preferences for linkage patterns that are, to a certain extent, independent of text, we do not want to ignore text. Both HMMs and CRFs are convenient in this regard as they provide a natural way to model both kinds of evidence. We model flow with state transition probabilities learned from a training corpus, and we model text with emission probabilities derived from the scores of learned language models. We first describe our HMM approach, and then we describe our related CRF approach.

We consider two HMM constructions: "sentences in states" (SIS) and "figures in states" (FIS). Our description is in terms of the SIS construction, but FIS can be understood by swapping 'sentence' with 'figure' in the description. Under the SIS construction, an article with J sentences and K figures has an HMM with $J+1$ states, $\{0\},\{1\},\{2\},\cdots,\{J\}$, and the length K observation sequence $(1,2,\cdots,K)$. State $\{j\}$ is associated with abstract sentence j, and the non-linked state $\{0\}$ is not associated with any sentence. We have a transition between every pair of states. Figure 7(a) shows an example HMM.

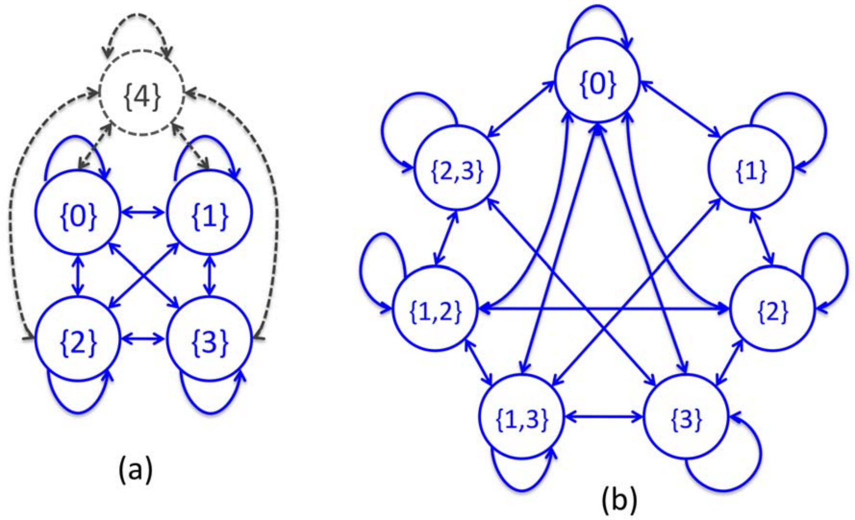

Figure 7. Example HMM (a) and CRF (b) state transition diagrams using the sentences-in-states construction. (a) States and transitions for the base HMM for a corpus where the maximum number of abstract sentences in an article (J^*) is 4. The states and transitions in sold blue are part of the derived HMM for an article with $J=3$ sentences. (b) CRF states and transitions for an article with $J=3$ sentences and where the maximum number of sentences per figure (M) is 2.
doi:10.1371/journal.pone.0039618.g007

We associate linkage-predictions with state-sequence paths. For example, for an article with $K=5$ figures and $J=3$ sentences, the path [{2},{0},{2},{3},{0}] asserts that Figure 1 links with Sentence 2, Figure 2 does not link to any sentence, Figure 3 also links with Sentence 2, Figure 4 links with Sentence 3, and Figure 5 does not link with any sentence. With the SIS construction, a single path can link a sentence to any number of figures (0 to K), while a figure can only link with 0 or 1 sentences. Our CRF approach relaxes this constraint and permits figures to link with multiple sentences.

As articles have different numbers of abstract sentences, their HMMs have different numbers of states. Our approach to this variation is to learn transition probabilities for a single base HMM structure with J^*+1 states, where J^* is the maximum number of abstract sentences in any article of the training corpus. Then, for an article with J abstract sentences, we construct a $J+1$-state HMM to predict linkages. The transition probabilities of the derived HMM come from the base structure, while the emission probabilities are based on language model scores.

Training the base HMM structure involves estimating the entries of its (J^*+1)-by-(J^*+1) transition matrix U. The value $U(j,j')$ is the probability for the transition from $\{j\}$ to $\{j'\}$. In other words, $U(j,j')$ is the probability of $j' \leftrightarrow k$ given $j \leftrightarrow (k-1)$, for all k. Here, we include unlinked figures in our notation by defining $0 \leftrightarrow k$ to mean that k is unlinked. We estimate U from the training corpus's transition counts matrix C, where $C(j,j')$ is the number of times that $j \leftrightarrow k$ and $j' \leftrightarrow (k+1)$ in the training set:

$$C(j,j') = \sum_{d}^{D} \sum_{k=2}^{K^d} \mathcal{L}^d(j,k-1)\mathcal{L}^d(j',k). \tag{15}$$

Here d indexes training set documents, and $\mathcal{L}^d$ is the linkage matrix for training document d. We set the $U(j,k)$ to their MAP estimate using Dirichlet priors with hyperparameters set to 1.0:

$$U(r,s) = \frac{C(r,s)+1}{J^* + \sum_t C(r,t)}. \tag{16}$$

We create the derived HMM with states $(\{0\},\{1\},\cdots,\{J\})$ by extracting the corresponding states and transitions between them from the base structure, and re-normalizing transition probabilities so that the sum of the outgoing probabilities from any state is 1.0. If a test-set article happens to have more abstract sentences than any article in the training corpus, we create its derived HMM by adding states to the base structure. Then, we also add transitions so that the derived HMM is fully connected, assign small probabilities to the new transitions, and re-normalize.

We now describe how we set the emission probabilities of the derived HMMs to model text. Since the observation for an article with K figures is the ordered sequence $(1,2,\cdots,K)$, state $\{j\}$ emits symbol k only if $j \leftrightarrow k$. We thus set $B(j,k)$, the emission probability for symbol k in state $\{j\}$, based on the textual coherence between abstract sentence j and figure k. While our language models, of course, are designed to do just this, setting the $B(j,k)$ directly from LM probabilities gives poor performance. The primary problem with this approach is that the LM probabilities are not well calibrated. As with other naive Bayes-like models, the posteriors of our LMs tend to be extreme, that is, very close to zero or one. Although models with uncalibrated probabilities often perform well in classification tasks [30], when such models are used as components within a larger model like an HMM predictions can be poor as inference in this case involves reasoning with many uncalibrated probabilities. Therefore, we represent the emission probabilities using Gaussian models of LM scores.

We learn one Gaussian model of LM scores for linked instances and another for non-linked instances. As this approach applies to the scores $\mathbf{S}^{LM}$ of any language model or the scores $\mathbf{S}^{LM+DM}$ of the combined language and distance model, for generality in our description we denote scores by $\mathbf{S}$ as the computations are the same in all cases. From the training corpus we calculate the sample mean and variance of $\mathbf{S}(j,k)$ over all linked (μ_L, σ_L^2) and non-linked (μ_N, σ_N^2) instances, and use these parameters to define Gaussian distributions for $\Pr(\mathbf{S}(j,k)|j \leftrightarrow k)$ and $\Pr(\mathbf{S}(j,k)|j \not\leftrightarrow k)$.

Let $\tilde{B}(j,k)$ be the joint probability under these models of all scores for figure k: $\mathbf{S}(1,k), \cdots, \mathbf{S}(J,k)$, given that it links only with sentence j, $j \geq 1$:

$$\tilde{B}(j,k) \overset{def}{=} \Pr\big(\mathrm{S}(1,k), \cdots, \mathrm{S}(J,k)\,|\,j \leftrightarrow k, j' \not\leftrightarrow k \text{ for } j \neq j'\big) \qquad (17)$$

$$= \mathbb{N}(\mathbf{S}(j,k), \mu_L, \sigma_L^2) \prod_{j' \neq j} \mathbb{N}(\mathbf{S}(j' \not\leftrightarrow k), \mu_N, \sigma_N^2) \qquad (18)$$

where $\mathbb{N}(x, \mu, \sigma^2)$ denotes the value of the probability density function for the Gaussian with parameters μ and σ^2 at x. Equation 18 assumes the elements of S are independent given $\mathcal{L}$. Similarly we define $\tilde{B}(0,k)$ for non-linked k:

$$\tilde{B}(0,k) \overset{def}{=} \Pr\big(\mathrm{S}(1,k), \cdots \mathrm{S}(J,k)\,|\,j' \not\leftrightarrow k \ \text{ for all } j'\big) \qquad (19)$$

$$= \prod_{j'} \mathbb{N}(S(j',k), \mu_N, \sigma_N^2) \qquad (20)$$

Lastly, we set the emission probabilities by normalizing the $\tilde{B}$'s.

$$B(j,k) = \frac{\tilde{B}(j,k)}{Z_{\{j\}}}, \qquad (21)$$

$$Z_{\{j\}} = \sum_k \tilde{B}(j,k). \qquad (22)$$

We define the HMM score for abstract sentence j and figure k, $\mathbf{S}^{HMM}(j,k)$, as the posterior probability that the state occupied on step k, π_k, is state $\{j\}$.

$$\mathbf{S}^{HMM}(j,k) \overset{def}{=} \Pr(j \leftrightarrow k) = \Pr(\pi_k = j|\mathbf{S}).$$

where the probability is with respect to the article's derived HMM for the standard observation sequence $(1, \cdots, K)$. We use the *posterior decoding* procedure [31] to compute the HMM scores.

Conditional Random Fields

Our HMM approach captures some properties of linkage features well, but fails to capture some others. Consider, for instance, the EdgesCrossed linkage feature and the transition $\{2\} \rightarrow \{1\}$. Whenever this transition is taken at step k, HMM semantics assert both $2 \leftrightarrow k$ and $1 \leftrightarrow (k+1)$, which induces a crossed edge in the linkage graph. Now, the HMM may learn a relatively small transition probability for $\{2\} \rightarrow \{1\}$, but only if other transitions from $\{2\}$ are more frequent in the training set. Standard HMM representations, however, provide no mechanism for generalization to transitions from other states using, for example, a common penalty for transitions that induce crossed edges. Such a general penalty is especially beneficial when learning transition probabilities, such as for $\{9\} \rightarrow \{8\}$, from less frequently visited states. Indeed, in our data set there are 98 total transitions from state $\{2\}$ but only 19 from state $\{9\}$. Conditional random fields [3] (CRFs), on the other hand, provide a mechanism for generalization through weights associated with a set of shared transition features.

Our CRF approach is similar in many respects to our HMM approach. Our CRFs also have SIS and FIS variants (we describe here the SIS variant), also associate linkage predictions with state sequence paths, and also generate a length K observation sequence $1, 2, \cdots, K$ for an article with K figures. Furthermore, like HMMs, the likelihood of a path through the model is proportional to the product of $K-1$ transition terms and K emission terms. There are, however, two key differences between our CRF and HMM methods. First, while each HMM state is associated with one or zero sentences, in CRFs we also have states associated with multiple sentences. These multi-sentence states enable us to link a figure with multiple sentences on a single path. Second, CRFs use a different representation transition affinity. While for HMMs the affinity of a transition is its transition probability, CRF transition affinity is given by a weighted sum of feature values. Sharing of features and weights enables information transfer across transitions.

A CRF for an article with J sentences has a state u for every subset of the sentences $\{1, 2, \cdots, J\}$ with size $\leq M$. In our experiments we set $M = 3$. (Figure 7(b) shows an example CRF with $M = 2$.) We use $\mathcal{S}(u)$ to refer to the set of sentences associated with state u so, $\mathcal{S}(u) \subset \{1, 2, \cdots, J\}$ and $\#(\mathcal{S}(u)) \leq M$ where $\#(\cdot)$ denotes set cardinality. The state sequence path for an article with K figures, $\vec{\pi} = \pi_1, \cdots, \pi_K$, asserts that figure k is linked with all abstract sentences in $\mathcal{S}(\pi_k)$. Thus, the number of abstract sentences linked with figure k (the degree of k), is $\#(\mathcal{S}(\pi_k)) \leq M$, and the number of figures linked with abstract sentence j is $\#(\{k|j \in \mathcal{S}(\pi_k)\})$ is $\leq K$. Since in the SIS construction, the degree of figure k is entirely determined from π_k we are able to readily encode figure degree properties in CRF transition features. On the other hand, as the degree of sentence j depends on the whole path, sentence degree properties are not as amenable to representation as transition features. One of the likely reasons that SIS representations outperform FIS representations is that the linkage feature FigureDegree is substantially more predictive of linkage than SentenceDegree (Table 2).

Our CRFs are parameterized by the transition feature weight vector $\vec{w} = w_1, \cdots, w_F$, where F is the number of transition features, and weight w_i is associated with transition feature f_i. The weight vectors are set during training. Given $\vec{w}$, the CRF probability of the path $\vec{\pi} = \pi_1, \pi_2, ..., \pi_K$ is proportional to the product of the start-state affinity $(\mathcal{Q}^s)$, K emission affinities $(\mathcal{Q}^e)$, and $K-1$ transition affinities $(\mathcal{Q}^t)$:

$$\Pr(\vec{\pi}|\vec{w}) \propto \left(\prod_{k=1}^{K} \mathcal{Q}^e(\pi_k, k)\right) \left(\mathcal{Q}^s(\pi_1|\vec{w}) \prod_{k=2}^{K} \mathcal{Q}^t(\pi_{k-1}, \pi_k|\vec{w})\right) (23)$$

Here, $\mathcal{Q}^s(\pi_1|\vec{w})$ is the affinity for starting in state π_1, $\mathcal{Q}^t(\pi_{k-1},\pi_k|\vec{w})$ is the affinity for the transition $\pi_{k-1} \rightarrow \pi_k$, and the emission affinity $\mathcal{Q}^e(\pi_k,k)$ gives the affinity for linking figure k with sentences $\mathcal{S}(\pi_k)$. The emission affinities represent the textual coherence of the implied linkages, and are defined similarly to the emission probabilities of our HMMs. Also, the emission affinities do not depend on $\vec{w}$, and so, as with HMM emission probabilities, they are not adjusted during CRF training.

Transition affinities. We now describe the transition features $\vec{f}(u,v)$ we use to represent the start-state and transition affinities. We represent the transition affinity for the transition from state u to v using the standard log-linear model:

$$\mathcal{Q}^t(u,v|\vec{w}) = \exp\left(\sum_i w_i f_i(u,v)\right), \tag{24}$$

where $f_i(u,v)$ is the value of feature i for $u\rightarrow v$. We have a similar representation for the start-state affinity:

$$\mathcal{Q}^s(v|\vec{w}) = \exp\left(\sum_i w_i f_i(0,v)\right), \tag{25}$$

where $f_i(0,v)$ denotes the value of feature i associated with starting in v. To simplify description of our features below, we define $\mathcal{S}(0) \overset{def}{=} \varnothing$.

We now describe our eight transition features. These features are closely related to the linkage features described above. However, as each transition only provides information on linkage of two neighboring figures, each feature $f_i(u,v)$ can only depend on two adjacent columns of the linkage matrix $\mathcal{L}$ (see Figure 6). We have a group of four binary features related to the number of sentences in the destination state v. The names of these features all begin with " FigureDegree" because the degree of figure k's vertex in the graph representation is equal to $\#(\mathcal{S}(\pi_k))$. Each of these features is a binary test on $\#(v)$, and for every state exactly one of these features is 1 and all other features are 0.

$$\text{FigureDegreeIs } 0(u\rightarrow v) = \left(\begin{array}{ll} 1 & \text{if } \#\ (\mathcal{S}(v)) = 0 \\ 0 & \quad \text{otherwise} \end{array}\right. .$$

$$\text{FigureDegreeIs } 1(u\rightarrow v) = \left(\begin{array}{ll} 1 & \text{if } \#\ (\mathcal{S}(v)) = 1 \\ 0 & \quad \text{otherwise} \end{array}\right. .$$

$$\text{FigureDegreeIs } 2(u\rightarrow v) = \left(\begin{array}{ll} 1 & \text{if } \#\ (\mathcal{S}(v)) = 2 \\ 0 & \quad \text{otherwise} \end{array}\right. .$$

$$\text{FigureDegreeIs } 3 \text{ Plus}(u\rightarrow v) = \left(\begin{array}{ll} 1 & \text{if } \#\ (\mathcal{S}(v)) \geq 3 \\ 0 & \quad \text{otherwise} \end{array}\right. .$$

Our next feature, $\text{NumEdgesCrossed}(u\rightarrow v)$, counts the number of crossed edges in the graph representation induced by the linkages implied by the transition. While this feature estimates the linkage feature EdgesCrossed it will not count crossed edges that require more information about $\mathcal{L}$ than what is implicit in the transition.

$$\text{NumEdgesCrossed}(u\rightarrow v) = \#\left(\{\ (j,j')\ |\ j\in\mathcal{S}(u), j'\in\mathcal{S}(v), j>j'\ \}\right).$$

The last group of transition features (PreviousFigure, PreviousSentence and PrevSentAndFig) count the number of occurrences of neighborhood linkage patterns. Recall that $\mathcal{S}(u)$ denotes the set of sentences associated with state u. Thus, a path that includes transition $u\rightarrow v$ asserts that some figure k is linked with all the sentences $\mathcal{S}(v)$ associated with the destination state v, and that the previous figure $k-1$ is linked with all the sentences $\mathcal{S}(u)$ associated with the source state u. $\text{PreviousFigure}(u\rightarrow v)$ is the count of the number of times a sentence links with both figure k and the previous figure $k-1$:

$$\text{PreviousFigure}(u\rightarrow v) = \#(\mathcal{S}(u)\cap\mathcal{S}(v)).$$

Similarly, $\text{PreviousSentence}(u\rightarrow v)$ is the count of the number of times figure k links with both sentence j and the previous sentence $j-1$:

$$\text{PreviousSentence}(u\rightarrow v) = \#(\{\ j\ |\ \{j-1,j\}\subseteq\mathcal{S}(v)).$$

Note that $\text{PreviousSentence}(u\rightarrow v)$ depends only on the sentences $\mathcal{S}(v)$ in the destination state. Although additional "previous sentence" counts can be inferred from linkages between figure $k-1$ and source state sentences $\mathcal{S}(u)$, to prevent double counting we do not count them on $u\rightarrow v$ because they get counted on the transition into u. We define the last neighborhood feature, $\text{PrevSentAndFig}(u\rightarrow v)$, to be the count of the number of times figure k links with sentence j while the previous figure $k-1$ links with the previous sentence $j-1$:

$$\text{PrevSentAndFig}(u\rightarrow v) = \#(\{\ j\ |\ j\in\mathcal{S}(u), j-1\in\mathcal{S}(v)\}).$$

Unlike $\text{NumEdgesCrossed}(u\rightarrow v)$, the values of the three neighborhood features are exact.

As an example, consider the transition $\{2\}\rightarrow\{1,3\}$ with $u=\{2\}$ and $v=\{1,3\}$. We have.

$$\text{FigureDegreeIs } 2(\{2\}\rightarrow\{1,3\}) = 1$$

$\text{NumEdgesCrossed}\ (\{2\}\rightarrow\{1,3\}) = 1$ (from $k\leftrightarrow 1$ and $[k-1]\leftrightarrow 2$),

$$\text{PreviousFigure}\ (\{2\}\rightarrow\{1,3\}) = 0$$

$$\text{PreviousSentence}\ (\{2\}\rightarrow\{1,3\}) = 0$$

and $\text{PrevSentAndFig}\ (\{2\}\rightarrow\{1,3\}) = 1$ (from $k\leftrightarrow 3$ and $[k-1]\leftrightarrow 2$).

Emission affinities. The emission affinity for symbol k (for Figure k) in state u, $\mathcal{Q}^e(u,k)$, is based on the text coherence for all implied linkages and non-linkages. Emission of the symbol k from state u implies that sentences associated with u, $\mathcal{S}(u)$, are linked with figure k and all other sentences are not linked with k. The matrix of CRF emission affinities $\mathcal{Q}^e$ is closely related to the B matrix of HMM emission probabilities:

$$\tilde{\mathcal{Q}}^e(u,k) \overset{def}{=} \Pr(\mathrm{S}(1,k),\cdots,\mathrm{S}(J,k)\,|\,j\leftrightarrow k \text{ for all} j\in\mathcal{S}(u), j' \not\leftrightarrow k \text{ for all} j'\notin\mathcal{S}(u)) \tag{26}$$

$$= \prod_{j\in\mathcal{S}(u)} \mathbb{N}(S(j,k),\mu_L,\sigma_L^2) \prod_{j'\notin\mathcal{S}(u)} \mathbb{N}(S(j',k),\mu_N,\sigma_N^2). \tag{27}$$

where $\mu_L, \sigma_L^2, \mu_N, \sigma_N^2$, are as defined in the HMM section. We set $\mathcal{Q}^e$ from $\tilde{\mathcal{Q}}^e$ by normalizing:

$$\mathcal{Q}^e(u,k) = \frac{\tilde{\mathcal{Q}}^e(u,k)}{Z_{\{u\}}} \tag{28}$$

$$Z_{\{u\}} = \sum_k \tilde{\mathcal{Q}}^e(u,k) \tag{29}$$

Prediction and learning. We use standard algorithms for learning weights and predicting linkages [3]. For learning we use gradient ascent to maximize the probability of training set state sequences. For prediction, we compute posterior distributions over $\Pr(\pi_k)$ using the forward and backward dynamic programming passes.

Performance Measures

Recall-precision, F1, and ROC curves. We compute precision (P), recall (R) and false-positive rate (FPR) for the linkage predictions of a set of sentence/figure instances:

$$P = \frac{TP}{TP+FP} \quad R = \frac{TP}{TP+FN} \quad FPR = \frac{FP}{FP+TN}. \tag{30}$$

TP, TN, FP, FN are the number of true positive, true negative, false positive, and false negative predictions, respectively, where a "positive" instance is linked and a "negative" instance is not linked. A recall-precision curve plots R vs P, while a receiver operating characteristic (ROC) curve plots FPR vs R. Points on these curves are calculated by varying the threshold on linked score that separates positive predictions from negative predictions. The area under an ROC curve (AROC) ranges from 0.0 to 1.0 where the AROC of a random guess classifier is equal to 0.5. The F1 score of a classifier is the geometric mean of R and P: F1 = 2RP/(R+P).

Author Contributions

Conceived and designed the experiments: JPB JMC SA DPO HY. Performed the experiments: JPB JMC SA. Analyzed the data: JPB JMC SA. Wrote the paper: JPB JMC SA DPO HY.

References

1. Yu H, Lee M (2006) Accessing bioscience images from abstract sentences. In: ISMB (Supplement of Bioinformatics). 547–556.
2. Rabiner LR (1989) A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE 77: 257–286.
3. Lafferty J, McCallum A, Pereira F (2001) Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proc. 18th International Conf. on Machine Learning. Morgan Kaufmann, San Francisco, CA, 282–289.
4. Lee M, Cimino J, Zhu HR, Sable C, Shanker V, et al. (2006) Beyond information retrieval–medical question answering. AMIA Annu Symp Proc: 469–473.
5. Müller HM, Kenny EE, Sternberg PW (2004) Textpresso: an ontology-based information retrieval and extraction system for biological literature. PLoS Biol 2: e309.
6. Smalheiser NR, Swanson DR (1998) Using ARROWSMITH: a computer-assisted approach to formulating and assessing scientific hypotheses. Comput Methods Programs Biomed 57: 149–153.
7. Rzhetsky A, Iossifov I, Koike T, Krauthammer M, Kra P, et al. (2004) Geneways: a system for extracting, analyzing, visualizing, and integrating molecular pathway data. Journal of Biomedical Informatics 37: 43–53.
8. Hoffmann R, Valencia A (2005) Implementing the iHOP concept for navigation of biomedical literature. In: ECCB/JBI (Supplement of Bioinformatics). p. 258. URL http://dx.doi.org/10.1093/bioinformatics/bti1142.
9. Harris MA, Clark J, Ireland A, Lomax J, Ashburner M, et al. (2004) The Gene Ontology (GO) database and informatics resource. Nucleic Acids Res 32: D258–261.
10. Blake JA, Richardson JE, Davisson MT, Eppig JT (1999) The Mouse Genome Database (MGD): genetic and genomic information about the laboratory mouse. the Mouse Genome Database Group. Nucleic Acids Research 27: 95–98.
11. Hamosh A, Scott AF, Amberger JS, Bocchini CA, McKusick VA (2005) Online mendelian inheritance in man (OMIM), a knowledgebase of human genes and genetic disorders. Nucleic Acids Research 33: 514–517.
12. Alfarano C, Andrade CE, Anthony K, Bahroos N, Bajec M, et al. (2005) The biomolecular interaction network database and related tools 2005 update. Nucleic Acids Research 33: 418–424.
13. Zweigenbaum P, Demner-Fushman D, Yu H, Cohen KB (2007) Frontiers of biomedical text mining: current progress. Briefings in Bioinformatics 8: 358–375.
14. Murphy RF, Kou Z, Hua J, Joffe M, Cohen WW (2004) Extracting and structuring subcellular location information from on-line journal articles: The subcellular location image finder. In: Proceedings of IASTED International Conference on Knowledge Sharing and Collaborative Engineering (KSCE-04), St. Thomas, US Virgin Islands. Morgan Kaufmann, 109–114.
15. Murphy RF, Velliste M, Yao J, Porreca G (2001) Searching online journals for uorescence microscope images depicting protein subcellular location patterns. In: IEEE Symposium on Bioinformatics and Bioengineering (BIBE). 119–128.
16. Murphy RF (2005) Cytomics and location proteomics: automated interpretation of subcellular patterns in uorescence microscope images. Cytometry A 67: 1–3.
17. Murphy RF (2005) Location proteomics: a systems approach to subcellular location. Biochem Soc Trans 33: 535–538.
18. Murphy RF (2004) Automated interpretation of protein subcellular location patterns: implications for early cancer detection and assessment. Ann N Y Acad Sci 1020: 124–131.
19. Rafkind B, Lee M, Chang SF, Yu H (2006) Exploring text and image features to classify images in bioscience literature. In: BioNLP '06: Proceedings of the Workshop on Linking Natural Language Processing and Biology. Morristown, NJ, USA: Association for Computational Linguistics, 73–80.
20. Shatkay H, Chen N, Blostein D (2006) Integrating image data into biomedical text categorization. In: ISMB (Supplement of Bioinformatics). 446–453. URL http://dx.doi.org/10.1093/bioinformatics/btl235.
21. Hearst MA, Divoli A, Ye J, Wooldridge MA (2007) Exploring the efficacy of caption search for bioscience journal search interfaces. In: BioNLP '07: Proceedings of the Workshop on BioNLP 2007. Morristown, NJ, USA: Association for Computational Linguistics, 73–80.
22. Hearst MA, Divoli A, Guturu H, Ksikes A, Nakov P, et al. (2007) Biotext search engine: beyond abstract search. Bioinformatics 23: 2196–2197.
23. Xu S, McCusker J, Krauthammer M (2008) Yale image finder (YIF): a new search engine for retrieving biomedical images. Bioinformatics 24: 1968–1970.
24. Yu H, Liu F, Ramesh BP (2010) Automatic figure ranking and user interfacing for intelligent figure search. PLoS ONE 5: e12983.
25. Dagan I, Marcus S, Markovitch S (1995) Contextual word similarity and estimation from sparse data. Computer Speech and Language 9: 123–152.
26. Jing H, McKeown KR (2000) Cut and paste based text summarization. In: Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 178–185.
27. Zhai C (2008) Statistical language models for information retrieval a critical review. Found Trends Inf Retr 2: 137–213.
28. Hiemstra D (2001) Using Language Models for Information Retrieval. Ph.D. thesis, University of Twente.
29. Ponte JM, Croft WB (1998) A language modeling approach to information retrieval. In: SIGIR. ACM, 275–281.
30. Domingos P, Pazzani M (1997) On the optimality of the simple Bayesian classifier under zero-one loss. Machine Learning 29: 103–130.
31. Durbin R, Eddy S, Krogh A, Mitchison G (1998) Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge, UK: Cambridge University Press.
32. Das S, Dixon J, Cho W (2003) Membrane-binding and activation mechanism of PTEN. Proceedings of the National Academy of Sciences 100: 7491–7496.