# Mining Text Networks: A Cross-Disciplinary Science

David Banks
Duke University

# 1. Introduction

Text networks arise in many situations:

- the Wikipedia is a network whose nodes are articles and whose edges are hyperlinks; each article contains text;
- citation networks;
- Internet webpages.

One would like to use information in the text to improve network models of connectivity, or growth and change.

Progress would enable one to find "holes" in the Wikipedia, discover overlooked references, improve recommender systems, and identify certain kinds of plagiarism.

A fair amount of previous work has been done in this area. The two main approaches involve:

- Natural language models
- Bag-of-words models.

The latter approach ignores semantic information: "I am not a crook" and "Am I not a crook" provide equivalent signal.

In contrast, natural language models attempt to include semantic information. In the context of text networks, there is a connection to the Semantic Web (cf. Allan Collins and Tim Berners-Lee), which attempts to provide hypertext metadata to provide "a web of data that can be processed directly and indirectly by machines" (Berners-Lee).

Natural language models are really hard. Oddly, the bag-of-words models are insanely successful for many purposes.

Presumably, the ideal in a natural language model would be to incorporate Chomskian deep structure as the core model, with surface structure elaborations pertinent to whatever language one is trying to model.

Deep structure models describe the process by which syntax creates rules that govern word order and sentence construction. The main idea is that deep structure describes the ways in which human brains are hardwired for language, and the surface structure are the more arbitrary conventions that distinguish, say, verb placement in English from verb placement in German.

It is not clear that deep structure really exists, of course, and it is not clear how to describe it. But computational linguists are busy and have made interesting progress.

Studies of Creole languages have added a lot of weight to the deep structure model (cf. Derek Bickerton, 1981). So have studies of language acquisition in early childhood (cf. Pinker, 1994).

**Note:** Personally, I'm not yet sold.

Achieving real text mining through a deep/surface structure model is a long ways off. It is nearly equivalent to the problem of AI.

Therefore, in practice, many researchers use $n$-grams. An **$n$-gram** is a sequence of $n$ words or word stems.

A **word stem** is a base word. The words "swam," "swum," "swim," "swimming" all map to the same base.

In trying to understand meaning, it is usually helpful to ignore tense, plurals, and other minor variations. There are sophisticated programs to do stemming, for multiple languages. Many are commercial; Snowball is a famous one.

Some stemmers have rules for stripping off suffixes. Others rely upon complex table look-ups. The trec studies at NIST have compared a number of different stemmers. It seems one can get 80% of the job done pretty easily, and then has to fight hard for every percentage point after that.

The point of looking at $n$-grams is to identify the probabilities of meaningful strings of words or roots.

For example, English has a window of about 8 to 9 before the Shannon entropy measure gets really high. This means that, after being told a specific word in a communication, the conditional probability of the eighth or ninth word after that is essentially the raw frequency of that word in common usage.

For example, if the first word is "how" then with high probabability the next word will be "are" or "can" or "is" or "will" or "do" or a handful of others. And the third word is, with fairly high probability, one of "you" or "we" or "I" or "one" or "my" or "your" or "Mom" and so forth. This ripple of excess probability flattens out to something close to baseline after about 8 or 9 words. (Obviously, this breaks down for nursery rhymes and other patterned speech.)

Abbott (2009) finds that dolphin $n$-grams flatten out at about 4.

There are a number of text-mining games one can play with $n$-grams. One strategy is model the matrix of transition probabilities that determine the probabilities that a given word follows another word, or follows at gap one, or gap two, and so forth.

This leads to Markov text generation.

From a network perspective, one can imagine that linked documents share common **topics**, and that the $n$-step Markov transition matrix for one topic is different from that of another. Then, depending on how one models topics, one can try to estimate the topic-specific transition matrices.

The differences between such matrices could flag the important differences in meaning and construction between documents on mathematics, documents on biology, and documents on sociology, say.

**Note:** I suspect math papers have a Shannon horizon greater than 9 words.

A second strategy for dealing with meaning in text is latent semantic indexing.

**Latent Semantic Indexing** is a procedure that addresses semantic problems of synonomy and polysemy by interpreting the meaning of words in the context of other words in the same document.

Synomyms are an issue for $n$-grams. Probability for the same "meaning" gets allocated across multiple sequences. But LSI can recognize synonyms:

- reduce the deficit by raising taxes on the wealthy
- reduce the deficit by raising taxes on job creators
- reduce the deficit by raising taxes on fat cats

lead to the phrases "wealthy," "job creators" and "fat cats" being nearby in term space.

**Note:** Since "job creators" and "fat cats" are actually two words, a little more pre-processing is needed to recognize their joint appearance as a single meaning.

Polysemy is harder; it requires disambiguations, and one wants to use only cues in the text, not domain knowledge, to do this.

For example, "Grateful Dead" can refer to a rock band or to a genre of German folktale. If the document includes the words "music" or "drugs" or "Haight-Ashbury" then the context suggests the former meaning. But if the document contains "woodcutter" or "coffin" or "magic goose" then the latter sense is implied.

To perform LSI, one does singular value decomposition (essentially a kind of factor analysis or principal components analysis) on a contingency table of text. This is sometimes called **correspondence analysis** (cf. Benzécri, 1973).

The method starts with a term-document matrix $\boldsymbol{X}$. The rows consist of all words in the corpus, the columns list all documents, and the cells contain counts for that word in the corresponding document.

Then one does some minor tranformations of the count, to normalize for the relative frequency of word within the document and the relative frequency of the word within the corpus.

The singular value transformation finds appropriate matrices $\boldsymbol{T}$ and $\boldsymbol{D}$ such that $\boldsymbol{X} = \boldsymbol{TSD}'$ where

- $\boldsymbol{S}$ is a diagonal matrix containing the singular values,
- $\boldsymbol{T}$ is the term matrix whose rows are eigenvectors that define the "term space",
- $\boldsymbol{D}$ is the document matrix whose columns are eigenvectors that define the "document space".

Usually it is good to truncate the $\boldsymbol{S}$ matrix.

The similarity of terms (synonymy) determines distance in the term space, and the similarity among the documents determines the amount of common content. So some documents with "Grateful Dead" will cluster in the region of document space that corresponds to music, and others will cluster in the folklore region.

**Note:** With tensor products, it might be possible to define term space, document space, and topic space?

# 2. Bags-of-Words

Rather remarkably, non-semantic methods have had remarkable success in text mining. These methods regard a document as a bag of words.

For text networks, one can look at the cross-entropy between documents. Suppose one document has the frequency distribution $f$ for its words, and another has the frequency distribtution $g$ for its content. Then the **cross-entropy** of

$$H(f,g) = -\sum_{x= \textbf{ all words}} f(x) \ln g(x).$$

Note that this is not symmetric in $f$ and $g$.

One would build a model in which documents with high cross-entropy have greater probability of being linked.

Topic models have become popular and important. These mostly rest on a statistical model for clustering, called the **Chinese Restaurant Process** (cf. Aldous, 1985, and Pitman, 1995).

The conceptual description is that a customer enters an empty restaurant, and picks a table at which to sit. Then a second customer enters, and with with a certain probability etiher joins the first customer or starts a new table. As future customers enter, they either join a previously chosen table, or start a new one.

The probability of joining others at a table increases with the number of people already present. It is a preferential attachment (Zipf's law) process.

This process yields a probability model for the partitioning (clustering) of the customers.

In the context of text, the tables are topics, the customers are documents, and each table has its own frequency distribution for words.

The Chinese restaurant process has some nice properties:

- It is exchangeable, in the sense that the order of the individuals entering does not affect the joint or marginal distributions.
- It is consistent, in that the probability distribution over the clusters with one person removed is the same as the probability distribution of the random partition if the process is run with one fewer customer.
- There are cool connections to the problem of partitioning the integers (i.e., in how many distinct ways can an integer be written as a sum of positive integers?).

There are several versions of the CRP, but the most standard has two parameters that control the probability of starting a new table and the rate at which a table increases in attractiveness as a function of the number already sitting at it.

Versions of the CRP are popular in statistics and machine learning. For text applications, hierarchical CRPs and CRPs with drift are important.

A related process is the **Indian Buffet Process**. (The IBP is to the beta process as the CRP is to the Dirichlet process.)

In an IBP, the story is that a customer goes to a buffet with an infinite number of dishes and selects $K_1$, where $K_1$ has a Poisson distribution with parameter $\lambda$.

After that, the $i$th customer chooses among the previously chosen dishes with probability $m_j/(i+1)$, where $m_j$ is the number of times that dish $j$ has been previously chosen. (So popular dishes have higher chance of selection.) Additionally, customer $i$ selects a Poisson number of previously unsampled dishes, where the Poisson probability is $\lambda/i$.

This approach allows one to do Bayesian nonparametrics with latent features. The Bayesian nonparametrics is the selection made by a specfic customer; the latent features are the dishes.

For topic modeling, hierarchical CRPs are hot. A document moves through a tree of restaurants, adding words to its bag as it goes.

The tables at the top restaurant select common words: a, the, of, etc.

Then the customers (documents) sitting at that table go off collectively to a new restaurant, sit at potentially different tables according to a new run of the CRP, and collect more specialized words according to parameters of the table at which they end up.

As one moves through this hierarchy, the bags of words become progressively specific, such that at the end the document (and those other documents that have selected the sequence of tables) might include specialized vocabulary: paramagnetic, permeability, molybdenum, Curie's.

IBPs can also be used to describe topic models. For example, you can regard the different dishes as frequency distributions of words. One dish might be mathematics, another astronomy, and a customer (document) who sampled both would be a paper on theoretical cosmology.

With these kinds of topic models, the strength of a link in document networks can be described in terms of

- the number of common tables at which the documents sat as they moved down the CRP hierarchy, or
- the number of IBP dishes that they both sampled.

Current research looks at how to impose correlation structure in the heirarchy, and how to model topic drift.

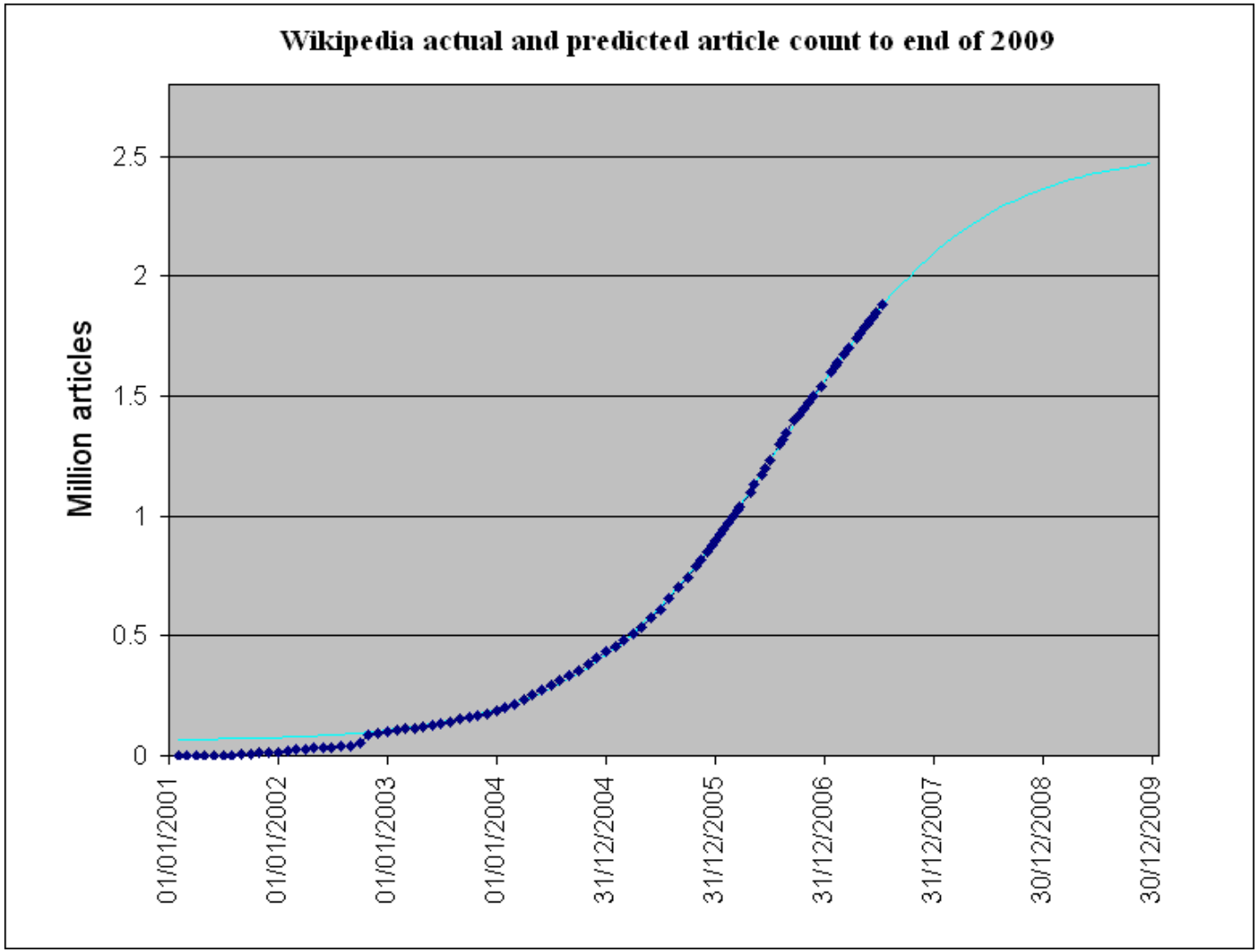
# A Case Study: The Wikipedia

The Wikipedia began in 1999, conceived by Jimbo Wales (with help from Larry Sanger), and went public on 1/15/2001. Its innovation is to encourage highly distributed collaborative construction and revision of content.

Jimbo Wales

Key facts about Wikipedia are:

- It uses wiki-ware to facilitate collaborations and the GNU Free Documentation License to avoid legal problems with ownership.
- Its quality and accuracy are enforced by the user community (and according to Nature, **438**, 900-901, it is more accurate than the Encyclopedia Britannica).
- It has internal and external links, and is a network model for the current state of human knowledge.
- **It has an open-access record of every change ever made, and who did it.**


Wikipedia actual and predicted article count to end of 2009
Million articles
2.5
2
1.5
1
0.5
0
01/01/2001
01/01/2002
01/01/2003
01/01/2004
31/12/2004
31/12/2005
31/12/2006
31/12/2007
30/12/2008
30/12/2009

From a dynamic modeling perspective, one wants to find a model that allows the formation of new nodes and new edges.

The key covariates are the bags of words corresponding to each of the articles. If the bags are very different, then the chance of an edge between them is small; and if the bags are similar, then then chance of an edge appearing between them is larger.

However, the logistic regression function one uses to estimate the probability of an edge should change as one moves around the Wikipedia network. The weights on "normal" and "distribution" might be high in the vicinity of the Statistics category, but low in the vicinity of the Ancient History. We are using a multi-task learning Bayesian elastic net—this can borrow strength from nearby articles.

To model the appearance of nodes, things are a little more complicated. If two nodes have bags that are very similar, then there may not be room for a new article between them. But if two nodes have bags that are very dissimilar, then there may be no sensible new entry that directly links to both.

Three questions of fundamental interest are:

- Can one find "holes" in the Wikipedia? That is, can one use recent history, latent semantic indexing, and local connectivity patterns to predict where new entries will appear in the Wikipedia network?
- Are there patterns in the local network structure? For example, is the local network around "homotopy" similar to the local network around "Henry VIII"? More generally, which parts of the Wikipedia noösphere have similar connectivity patterns and which regions are different?
- Are there growth and/or evolution patterns in the Wikipedia network that lend themselves to automation? For example, can one identify cases where a disambiguation page is needed to differentiate among distinct concepts? Similarly, can one mine the structure of Wikipedia to identify opportunities for linkages between pages, in a manner similar to the the models for triad completion used in social networks?

The challenges to building a dynamic network model for the Wikipedia are substantial:

- The covariates for the nodes are textual; this brings in topic models or latent semantic indexing or cross-entropy.
- The Curse of Dimensionality: each word or stem is a potential covariate, and inference become more difficult in high dimensions.
- The Wikipedia is very large; extracting useful data files is a computational obstacle.
- One anticipates that distance metrics are local; covariates (words) that are useful in predicting relationships among articles change (but change slowly) as one moves around the Wikipedia network.
- There is great interest in predicting the appearance of new nodes, whereas in most network problems, the focus is on predicting new edges.
- The dynamic behavior in the Wikipedia changes over time. For example, the number of new entries in the topic area Statistics varies by year. The rate of new articles peaked in 2006 , and it has grown more slowly since then.

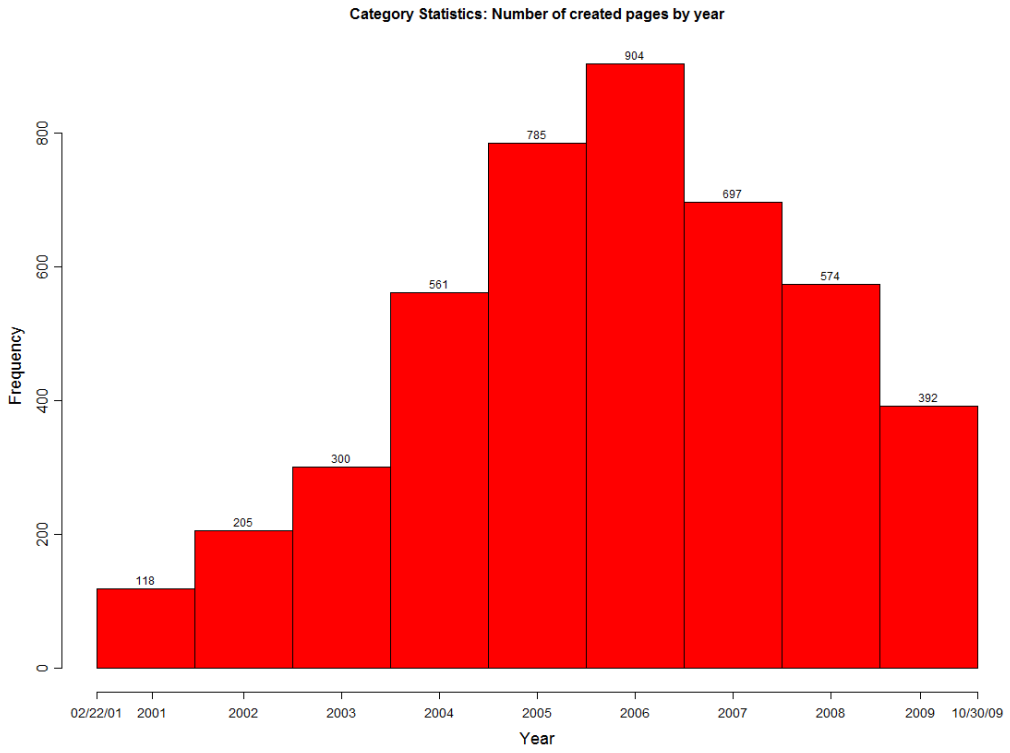



Figure 1: A barchart of the number of new articles in the topic category Statistics that were created in each year.

Consider the subtopic Continuous Distributions within the topic Statistics, which contains 96 articles (as of November 1, 2009).

The following figure shows several aspects of this region of the Wikipedia.

- The size of the circle reflects the betweenness centrality of the article; the article on the Normal Distribution has the highest value.
- The color indicates the in-degree of the article: at 48 links, the purple circle for the Normal Distribution is the largest value; close after that, the pink circles for Chi-Squared Distribution, Gamma Distribution, and Student's t-Distribution are prominent, and so forth.
- The connectivity pattern is shown by the links (and the direction of the link, if one looks closely to see the arrowheads). Note that more than 30 articles are linked to only one other entry.

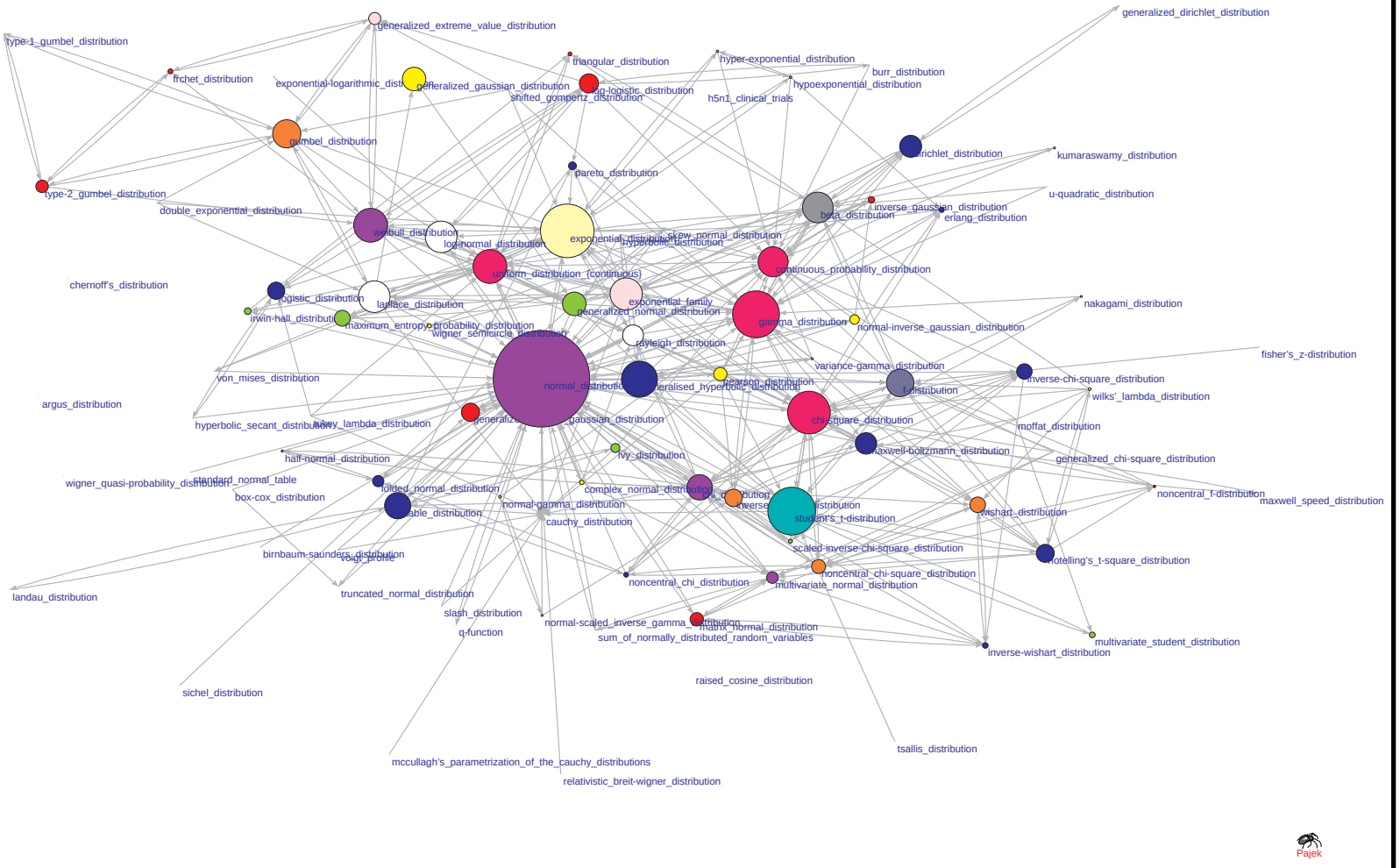



Figure 2: Connectivity and betweenness-centrality for the subtopic Continuous Distributions within the topic Statistics.

Since there are more covariates (words) than there are observations (edges and non-edges), modeling is hard. But one can borrow strength from nearby artcles. Logistic regressions that predict edges for nearby articles should depend on similar covariates which are given similar weights. This enables a **multi-task learning** approach.

When multi-task learning was applied to the Continuous Distribution region of the Wikipedia, it found 1034 words that were significantly useful in predicting edge-formation. One such word was "lambda". It appears in 23 articles, and is significant for 11 of them.

The following figure indicates the articles for which "lambda" was significant in red. The articles in the Continuous Distribution subtopic which receive links from a red circle but for which the word is not significant are shown in green. Articles in the topic Statistics, but not in the subtopic Continuous Distributions, that receive links from a red node, are shown in yellow.

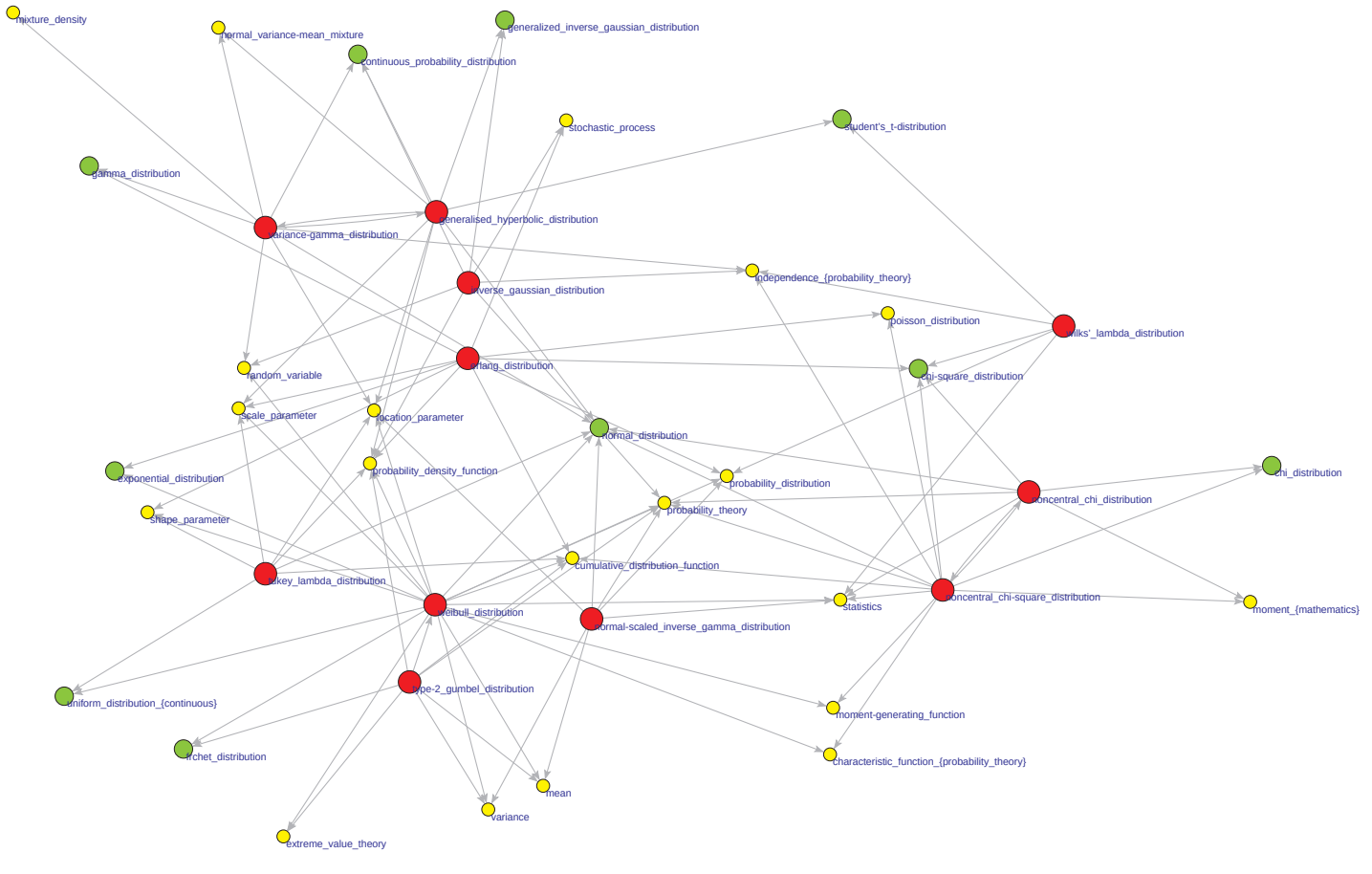



Figure 3: Articles for which "lambda" is an important predictor and connections to other articles in the Continuous Distribution subtopic and Statistics topic.

Note that the 11 articles for which the word is significant lie in essentially three clusters, so the strategy of borrowing strength seems to have been sensible.

(This figure is a simplified visualization, since it does not display links among the green and yellow circles; when those are included, the tightness of the clusters is stronger, although the figure is more cluttered.)

This all very exploratory, of course. A great deal more could be done to apply some of the other ideas described in this talk to the Wikipedia problem.

Dave Blei did a topic model of Wikipedia 3.3 million articles, and found about 900 topics. Would a graph partitioning algorithm find similar structure?