Part 1: Text Classification

Data	Model	Text	Task	Accuracy	Precision	Recall	F1-score
All train data	RNN	Preprocessing1	InfoTheory	94.86%	0.93	0.89	0.91
			CompVis	95.64%	0.94	0.83	0.87
			Math	86.60%	0.86	0.81	0.83
		Preprocessing2	InfoTheory	94.40%	0.9	0.91	0.91
			CompVis	95.86%	0.92	0.86	0.89
			Math	87.01%	0.86	0.82	0.84
	LR	Preprocessing1	InfoTheory	94.80%	0.94	0.88	0.91
			CompVis	96.06%	0.96	0.83	0.88
			Math	87.23%	0.86	0.83	0.84
		Preprocessing2	InfoTheory	94.74%	0.94	0.88	0.91
			CompVis	95.85%	0.96	0.82	0.88
			Math	87.22%	0.86	0.83	0.84
First 1000 rows	RNN	Preprocessing1	InfoTheory	81.62%	0.41	0.5	0.45
			CompVis	89.06%	0.45	0.5	0.47
			Math	69.86%	0.35	0.5	0.41
		Preprocessing2	InfoTheory	81.62%	0.41	0.5	0.45
			CompVis	89.06%	0.45	0.4	0.47
			Math	69.86%	0.35	0.5	0.41
	LR	Preprocessing1	InfoTheory	81.62%	0.41	0.5	0.45
			CompVis	89.06%	0.45	0.5	0.47
			Math	69.86%	0.35	0.5	0.41
		Preprocessing2	InfoTheory	81.62%	0.41	0.5	0.45
			CompVis	89.06%	0.45	0.5	0.47
			Math	69.86%	0.35	0.5	0.41

Table 1: Heat map of 24 results

text - preprocessing 1：[to lower, remove numbers, remove stop words, stemming]

text - preprocessing 2：[to lower, remove numbers, remove stop words, lemmatisation, remove rare tokens]

In the Heat Map of Table 1, accuracy is the only column I used all to calculate, while precision, recall and F1-score are too low for the first 1000 rows, so those data are not necessary to be considered. The range in these three scores which were using all data is the range of these heat maps.

It can be found that the F1-score of both models is very low when only using the first 1000 rows of data are used for training, and the accuracy of the RNN model is the same as that of the Logistical regression model and the performance of models using the different text-preprocessing methods are the same.

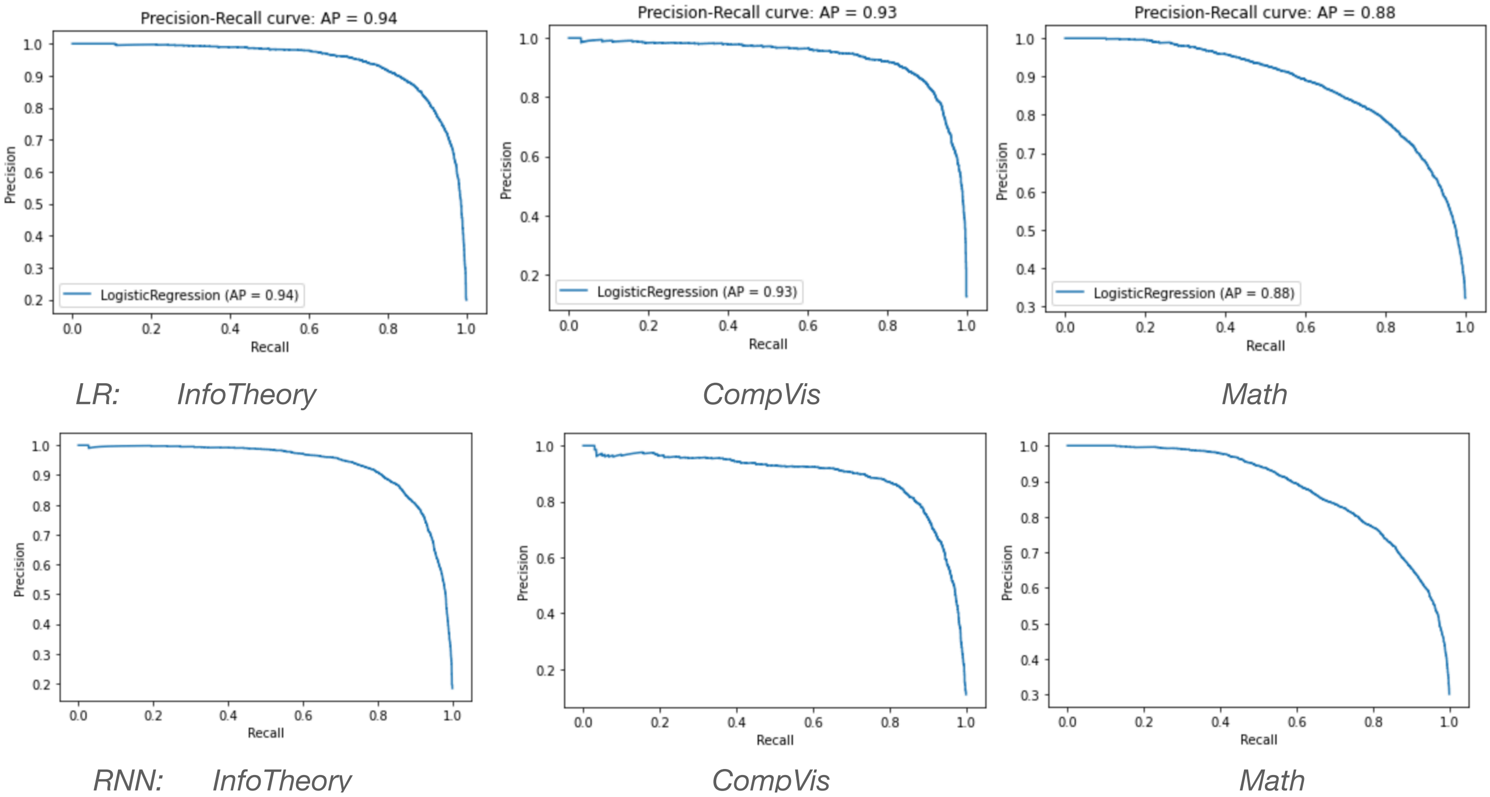
1. How well did the two algorithms work, when and why?

The accuracy value of both algorithms is reasonably high when classification InfoTheory and ComVis task, however, it is worth mentioning that Math has the lowest prediction rate no matter using all the data or 1000 rows of data. According to my guess, this is because Math related articles are usually designed in multiple fields and use words from this field, this makes the model harder to classification the article. Also, there are many symbols or equations in a Math-related article, this may affect the classification.

2. Which text pre-processing worked better, when and why?

By comparing the preprocessing 1 and 2 methods we can find that stemming and lemmatisation results are same, so my two preconditioning methods should have little effect on the model's predictions. This is because the meaning of most words is determined by their roots, and the effects of stemming and lemmatisation on those words which highly related to the category will be similar.

3. What insights do the various metrics and plots give you?


Precision-Recall curve: AP = 0.94

Precision-Recall curve: AP = 0.93

Precision-Recall curve: AP = 0.88

LR: InfoTheory *CompVis* *Math*

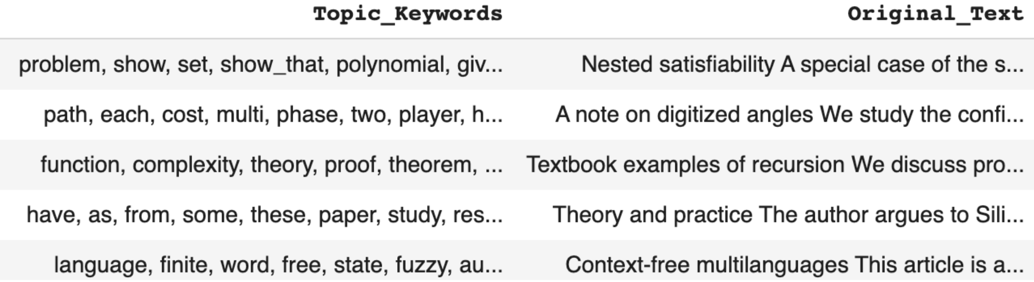
RNN: InfoTheory *CompVis* *Math*

Figure 1: Precision-recall curve of two model (prep 1, all data)

The accuracy and F1-score of these two algorithms in training all data sets is similar, and the accuracies are no more than 1% different. Both the InfoTheory and CompVis models performed well, reaching around 95% predictions and both had F1-scores above 0.87. According to Figure 1, it can be seen that the area enclosed by the PR curve of the two models is basically the same, which also indicates that the prediction performance of the two models is similar.

Part 2: Topic Modelling

1. Present what sorts of groupings there are about articles
What sorts of topics do you see?



Topic_Keywords	Original_Text
problem, show, set, show_that, polynomial, giv...	Nested satisfiability A special case of the s...
path, each, cost, multi, phase, two, player, h...	A note on digitized angles We study the confi...
function, complexity, theory, proof, theorem, ...	Textbook examples of recursion We discuss pro...
have, as, from, some, these, paper, study, res...	Theory and practice The author argues to Sili...
language, finite, word, free, state, fuzzy, au...	Context-free multilanguages This article is a...

Figure 2: topics and the article

Are all top topic words comprehensible sets of words?

No, not all top topic words are comprehensible, such as there are a group of topic keywords are: 'sentence, by, semantic, model, have, one, system, as, which, or….' These words were mainly stop-words, and we could not understand the topic they described, so we had better remove stop-words when we did topic modelling.

Perhaps find some articles that are examples and use them to illustrate key topics

In the Figure 2.

2. Describe how the topic modelling presents this and any advantages or shortcomings of topic modelling for the role in 1

We need to turn those text into vectors. First, we tokenise those words in the document and store them, which we call Bag-of-words, then we store the word bags into a dictionary, and the dictionary index is used to create a vector for each document and to indicate the frequency of the corresponding word. These data can then be put into the LDA model for calculation.
LDA is based on the Bayesian model, which means, the posterior distribution is obtained according to the prior distribution and data, and then used as the prior distribution of the next cycle.

Firstly, we determine the number of topics K, and then we get the Dirichlet distribution with N document topics, and the corresponding topic numbers present multinomial distribution, which forms the Dirichlet-Multi conjugate. Then we can use the Bayesian inference method to get the posterior distribution of document topics based on the Dirichlet distribution.
Similarly, for the distribution of topics and words, we can get the Dirichlet distribution of M topics and words, and the corresponding multinomial distribution of data with m topic numbers. Document topic distribution and subject word distribution are independent of each other because topic generation words do not depend on a specific document. Finally, the subject words are obtained by sampling from the word vector.

Advantages:

LDA is an unsupervised learning algorithm, which can train the model by given k, the number of topic we want. So the LDA algorithm can find words to describe each topic, no matter what kind of data we pass to it.

Weakness:

One of the disadvantages of LDA is that the topics it finds may be overlapping, and we may find many of the same words under multiple topics. In addition, the number of subjects we want for the K we give to LDA should not be too large. If the K is too large, it will lead to overfitting and increase the confusion between subjects.

3. To explain how your two configurations and data set sizes (1000, 20000)

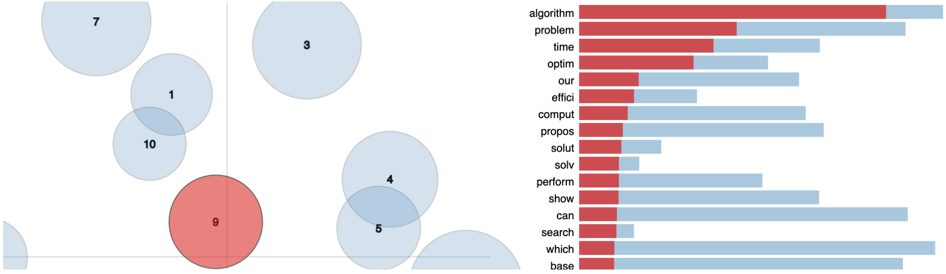


Figure 3:1000 and k=10

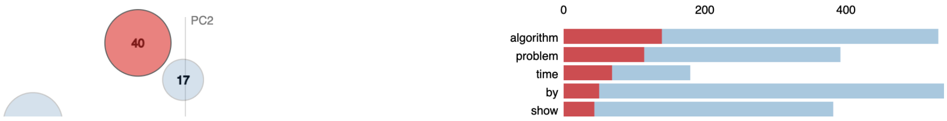


Figure 4: 1000 and k=40

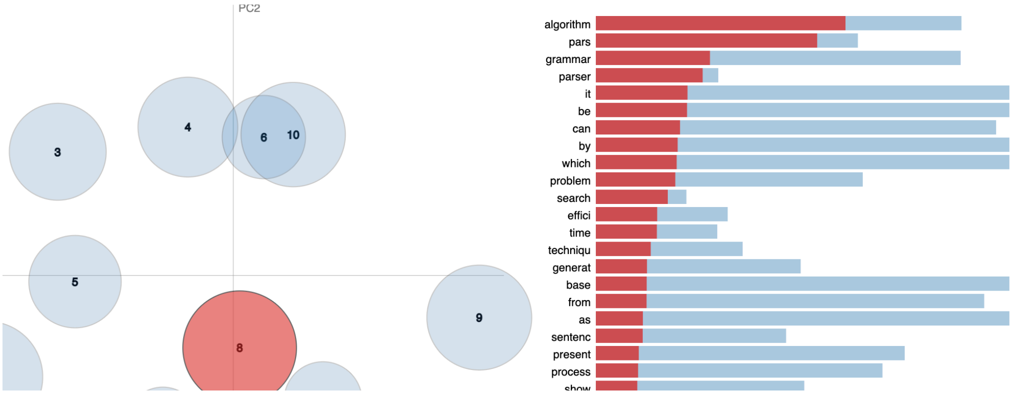


Figure 5: 20000 and k=10

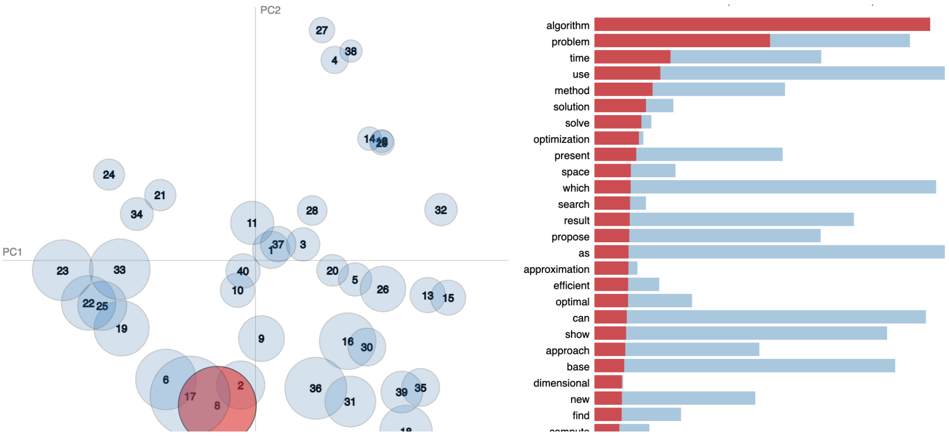


Figure 4:20000 and k=40

compare:

Compare k=10 and k=40, we can understand all the 10 topics according to their subject words when k=10, but we can't understand some topics words when k=40. This is one of the disadvantages I mentioned earlier, especially if we only have 1000 documents and k=40, It is difficult for the LDA to extract 40 topics from so little data.

Bigram and no bigram, we can see that basically all the subject words are one word, and the proportion of bigram in the document is too small, so we don't see the bigram in the subject words.