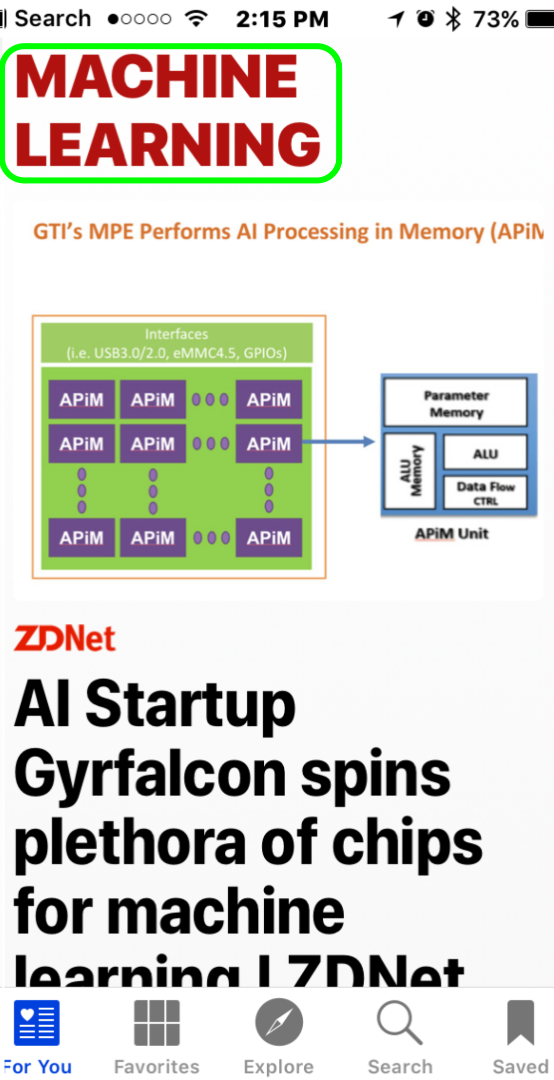
Chromatic News Article Recommender

Jayant Shelke
Mica Eldridge

Introduction

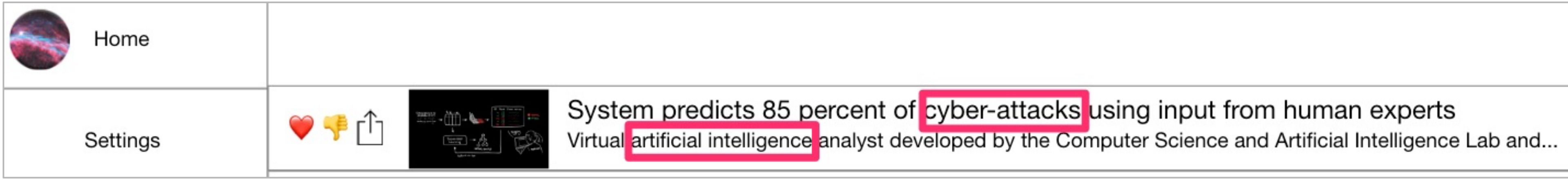
- Problems with existing news services
  - Rigid categories that don’t take inter-categorical interests into account
  - Interest categories that weren’t accounted for during manual curation of explicit categories
  - Limited to a specific set of news sources



Solution

- Solution
  - Automatically detect inter-categorical interests
  - Adapt to never-seen categories without manual work
  - Works for **any** text document, regardless of source

Sample suggestion that includes two interests: **Machine Learning and Information Security**:



Datasets

- Training Doc2vec and LDA: One Week of Global News Feeds dataset
  - 1.4 million article urls
  - Downloaded 640,500 of the corresponding articles using Python out of which 263,166 were used.
- Training 5-class classifier on LDA:
  - BBC Dataset
  - 2,225 documents
  - Five class labels: business, entertainment, politics, sport, tech
- User interest dataset
  - Found newsletter archives, then wrote loader to download the article texts referenced by the newsletters in that archive
  - Positive (upvote) dataset: Daniel Miessler's Unsupervised Learning newsletter
  - Negative (downvote) dataset: Casual Spectator Sports newsletter

BBC Dataset

Relatively small dataset when considering deep learning

Sample counts:

	Total	Train	Test	Percent test
Business	510	408	102	0.2
Sport	511	409	102	0.2
Politics	417	333	84	0.2
Entertainment	386	309	77	0.2
Tech	401	321	80	0.2

Sample article (business): Ad sales boost Time Warner profit; Quarterly profits at US media giant TimeWarner jumped 76% to $1.13bn (£600m) for the three months to December, from $639m year-earlier…

User interest dataset

- Very small dataset when considering deep learning
- Positive (upvote) dataset: Daniel Miessler’s Unsupervised Learning newsletter: contains anything of interest to Daniel Miessler, on topics such as Machine Learning, Information Security, Technology, self-driving vehicles, and IoT
- Negative (downvote) dataset: Casual Spectator Sports newsletter: topics regarding sports such as player information, game summaries, team politics, and injuries

Downloaded in total

Newsletter Archive Name	Article Count
Unsupervised Learning Daniel Miessler	828
Casual Spectator Sports	160
nathan.ai newsletter By Nathan Benaich	1427
Any Water Sports	71

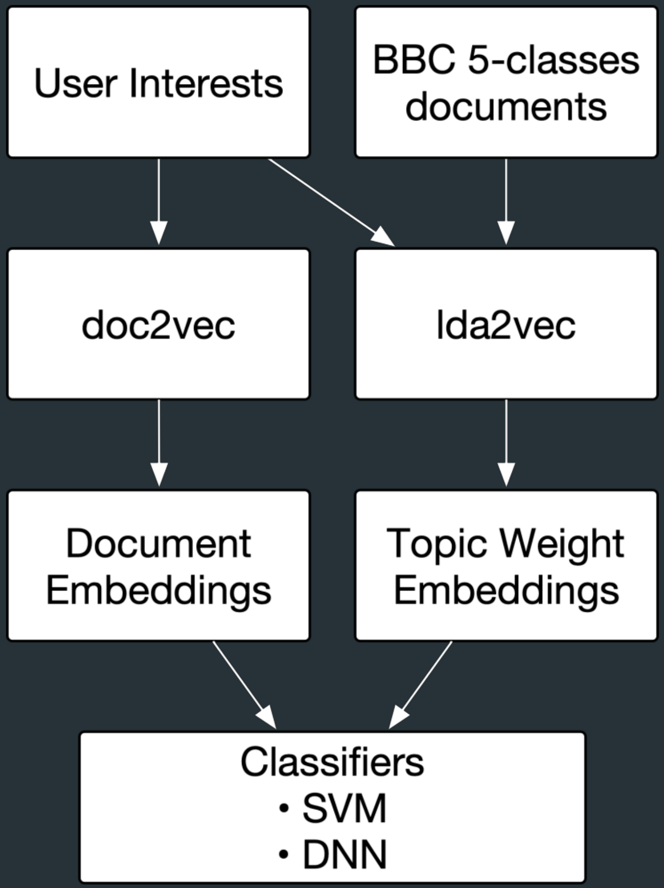
(4 rows)

Used for Machine Learning dataset

Label	Train	Test
Positive	72	18
Negative	72	18

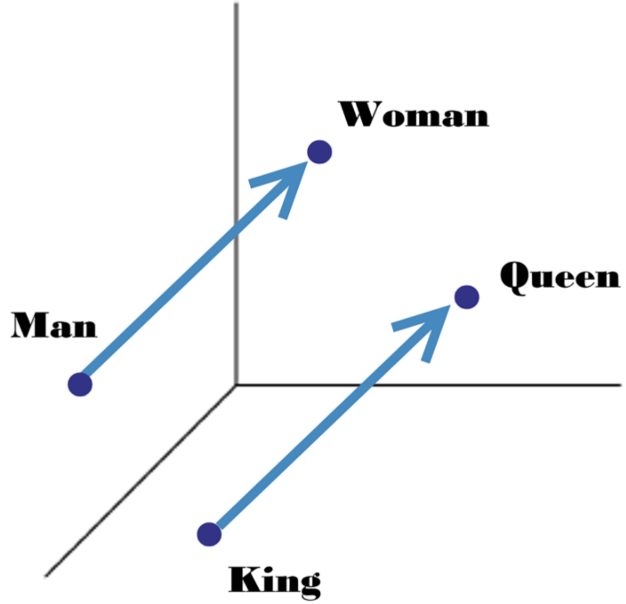
Algorithms

- Unsupervised
  - Doc2vec
  - LDA
- Supervised
  - Support Vector Classifiers
  - Keras Deep Neural Networks



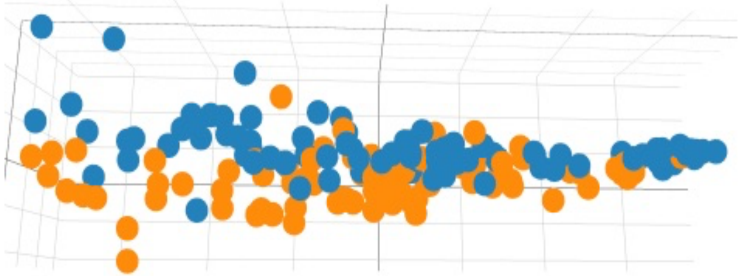
Background: Word2vec

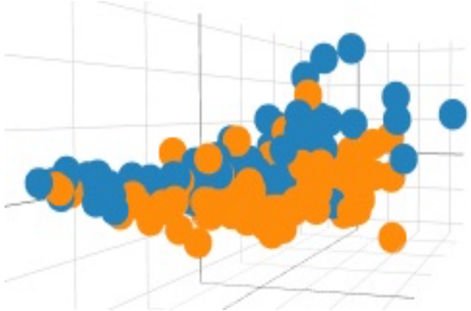
- W("woman")−W("man") ≃ W("queen")−W("king")
- https://www.oreilly.com/learning/capturing-semantic-meanings-using-deep-learning



Unsupervised: Doc2vec: User Interests 3D Plot

- Input document is a list of token strings, output is 100-entry float vector
- Two more-similar documents have closer embedding vectors than two less-similar documents
- Document embeddings of the two classes, Unsupervised Learning articles and Casual Spectator sports articles, reduced from 100 dimensions to 3 using PCA for 3D plotting:





Unsupervised: Doc2vec

- 229,410 documents from One Week of Global News Feeds dataset and User Interests dataset (so it could learn all the vocabulary)
- 19 training iterations, 2.5 minutes per iteration, approximately 47.5 minutes
- Sample document:

  “About a year before its scheduled official launch as the newest member of Washington State Ferries’ fleet, the superstructure...”

Unsupervised: LDA

Based on **Latent Dirichlet Allocation**

All topic models are based on the same basic assumption:

- each **document** consists of a mixture of ***topics***, and
- each ***topic*** consists of a collection of **words**.

What is a Topic?
Topic models are built around the idea that the semantics of our document are actually being governed by some hidden, or "latent" variables that we are not observing. Thus, the goal of topic modelling is to uncover these latent variables -- TOPICS -- that shape the meaning of our document and corpus.

Expected Output from LDA for a document:
{Topic1 : 60%, Topic2 : 35%, Topic3: 3%, Topic4 : 2%}

Unsupervised: LDA - Steps

LDA Steps:

- Read all Documents to be used for training.
- Clean the documents
  - Remove documents not in English
  - Remove stop words for keeping words that have better relevance.
  - Count word frequencies and remove the words which occur only once.
- Create dictionary of words - Collection of {Id:Token, Id:Token} pairs where Id is integer and token is word.
- Create corpus from this dictionary which is a vector of {Id: float, Id:float }
- Train the LDA model with this corpus, id2token pairs, num_topics, epochs, update_after.
- For a new document, pass the dictionary.doc2bow(document.SplitIntoWords()) output to the LDA model.
- This output is passed to LDA to get topic vectors.

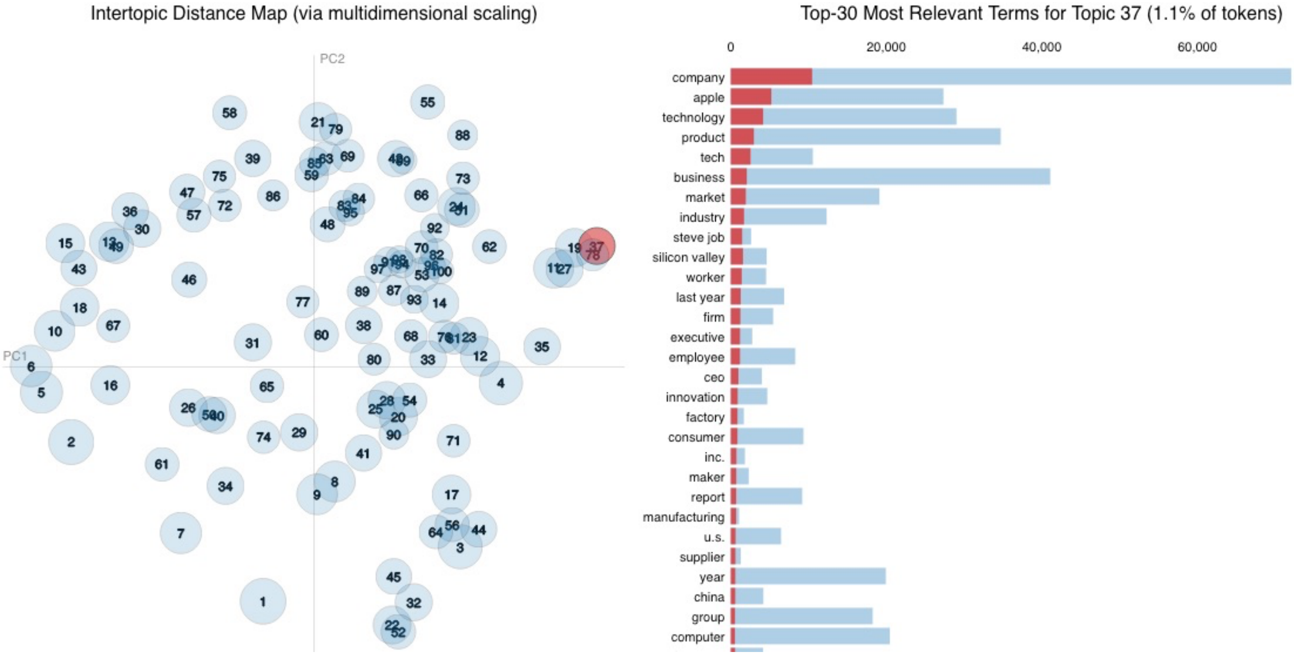
Hyper Parameters:

Num_topics : Total topics to be recognized in the learning.

Alpha : number of topic to be output in the topic mixture. Higher alpha - more mixture of topics

Beta : numbers of words that should contribute in the topic decision. Higher beta - more words in the topic.

Unsupervised: LDA



Article:
"The communications giant is expecting a windfall of $20bn in savings from **Trump's** tax reforms, but has closed 44 call centers since 2011…"

Detected Topics:

- support 0.221
- allegiance 0.099
- staff 0.091
- interview 0.065
- …
- **trumps** 0.014
- …

https://nbviewer.jupyter.org/github/bmabey/hacker_news_topic_modelling/blob/master/HN%20Topic%20Model%20Talk.ipynb#topic=37&lambda=1&term=
https://multithreaded.stitchfix.com/blog/2016/05/27/lda2vec/#topic=38&lambda=1&term=

LDA output for Keras training

Average number of topics per article: 16.91
Total number of topics: 1,000

- Document 1
  - support 0.221
  - allegiance 0.099
  - staff 0
  - interview 0.065
- Document 2
  - support 0
  - allegiance 0.099
  - staff 0.091
  - interview 0

Converted to matrices, fixed-length vectors for each document:

X.shape, y.shape
(894, 1000), (894,)

User Interests: Support Vector Classifiers

Doc2vec

Label	Train	Test
Positive	72	18
Negative	72	18

Model	Train Accuracy	Test Accuracy
LinearSVC	0.993056	0.805556
SVC	0.812500	0.833333
NuSVC	0.951389	0.861111

LDA

Label	Train	Test
Positive	603	151
Negative	112	28

Test Accuracies:

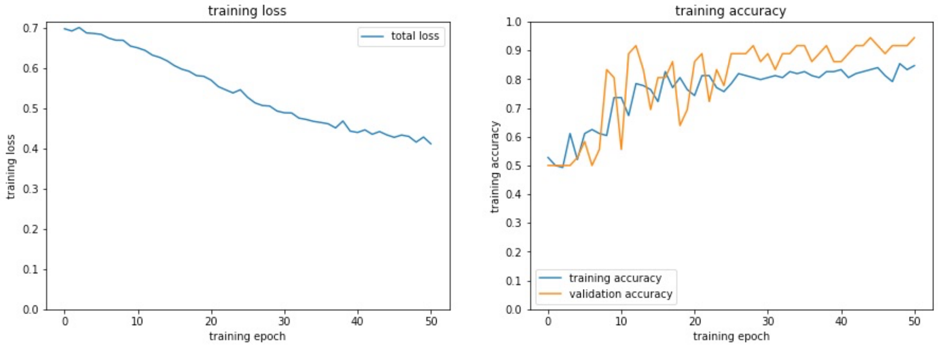
Model	Without mean normalization	With mean normalization
LinearSVC	0.966	0.950
OneClassSVM	0.642	0.553
NuSVC nu=0.1	0.994	0.972
NuSVC nu=0.2	0.966	0.950
NuSVC nu=0.3	0.927	0.905

User Interests:
Keras Deep Neural Networks

Doc2vec

Label	Train	Test
Positive	72	18
Negative	72	18

```
testing accuracy   : 0.854%
validation accuracy: 0.889%
```



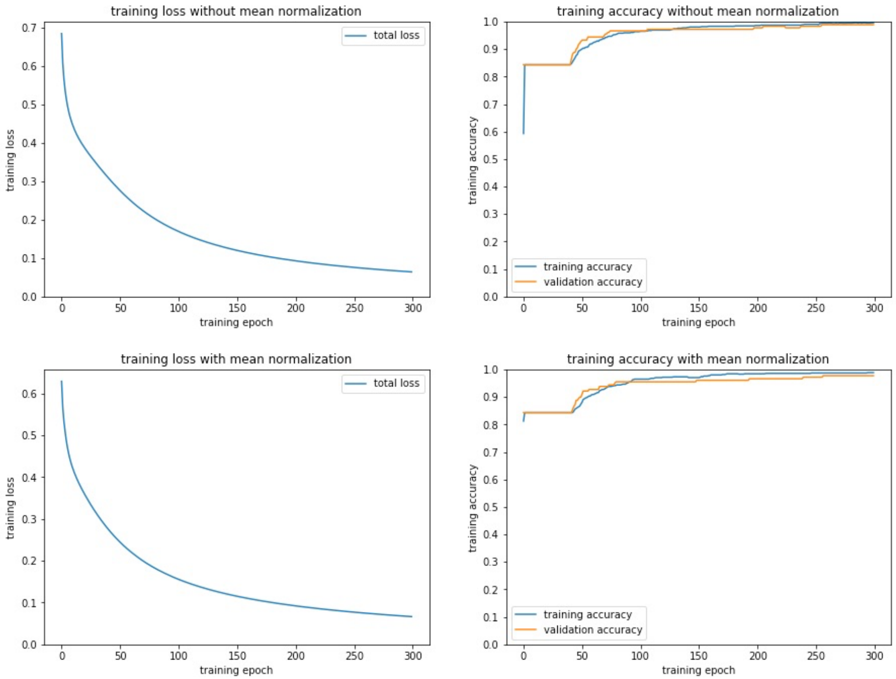
User Interests:
Keras Deep Neural Networks

LDA

Label	Train	Test
Positive	603	151
Negative	112	28

```
testing accuracy   : 0.989%
validation accuracy: 0.978%
```
<= Without mean normalization

```
testing accuracy   : 0.989%
validation accuracy: 0.978%
```
<= With mean normalization



Reviewing Results

Where did it initially fail?

- Non-article documents: 404s, “sign in to see”, “tweet this”
- Empty documents (null string)

Keras Deep Neural Networks

Architectures after some experimentation:

Doc2vec

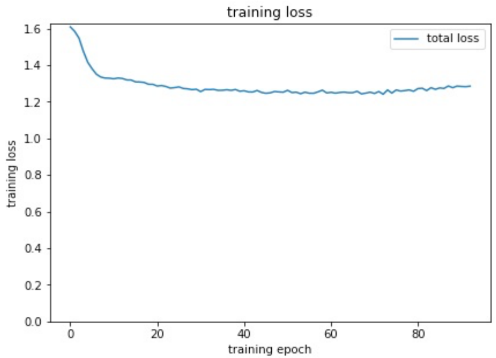
- 10, relu
- 30, sigmoid
- 30, sigmoid
- 1, sigmoid
- Optimizer: Adagrad
- Loss: Binary crossentropy

LDA

- 10, relu
- 10, relu
- 10, relu
- 5, softmax
- Optimizer: Rmsprop
- Loss: Categorical crossentropy

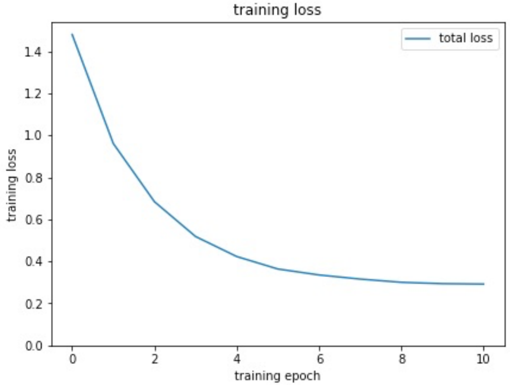
BBC 5-class Topic Classification: LDA

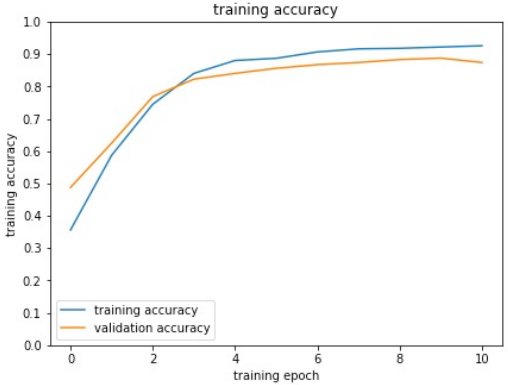
Trained on 31,000 topics




```
testing accuracy   : 0.935%
validation accuracy: 0.888%
```

Trained on 1,000 topics





LDA 5-fold Cross Validation:

User Interests, 5-fold CV: Test accuracies:
LinearSVC : 0.95 (+/- 0.004)
OneClassSVM : 0.77 (+/- 0.183)
NuSVC nu=0.1: 0.97 (+/- 0.006)
NuSVC nu=0.2: 0.94 (+/- 0.009)
NuSVC nu=0.3: 0.91 (+/- 0.010)
Keras : 0.93 (+/- 0.042)

5-class classification:

Keras, Categorical accuracy: 0.924±0.017

Conclusions

- Relatively high accuracy with relatively small dataset
- Combined unsupervised methods with supervised methods can have good results.