Toxic Comment NLP

Kaige Gao, Qiyuan Wang, Yijia Wei

Introduction

Decisions to be Impacted:

- Comment censorship
- Comment filtering

Business Value:

- Create a positive, inclusive, and friendly online environment
- Better user experience
- User retention rate
- Brand reputation
- Avoid potential legal disputes

Why we care about this project:

- Toxic content may cause emotional distress, discrimination spread, and reputation damage.

Data Asset Description

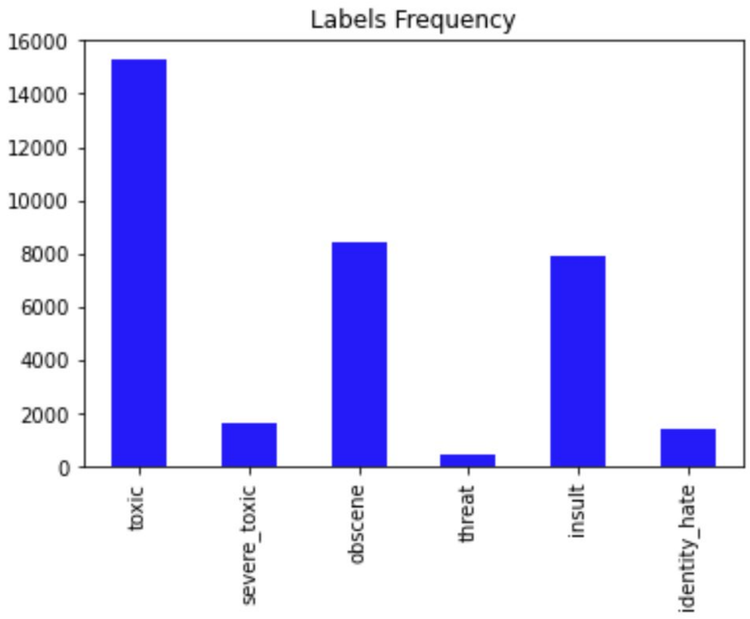
Wikipedia comments which have been labeled by human raters for toxic behavior.

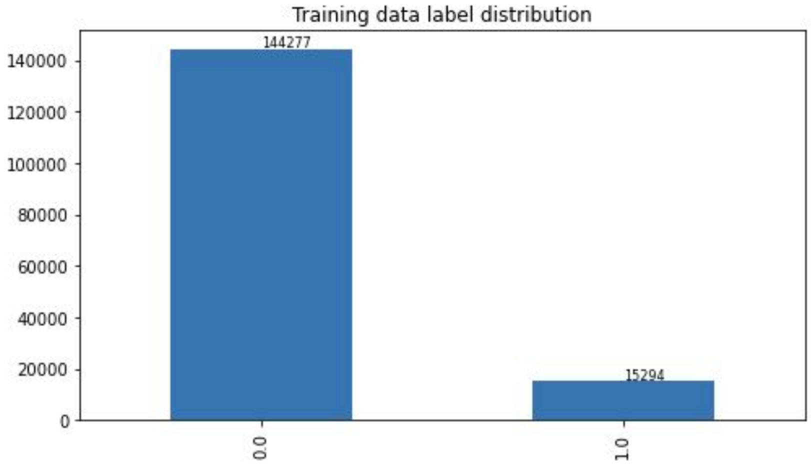
Sample size: 159,571

Feature:

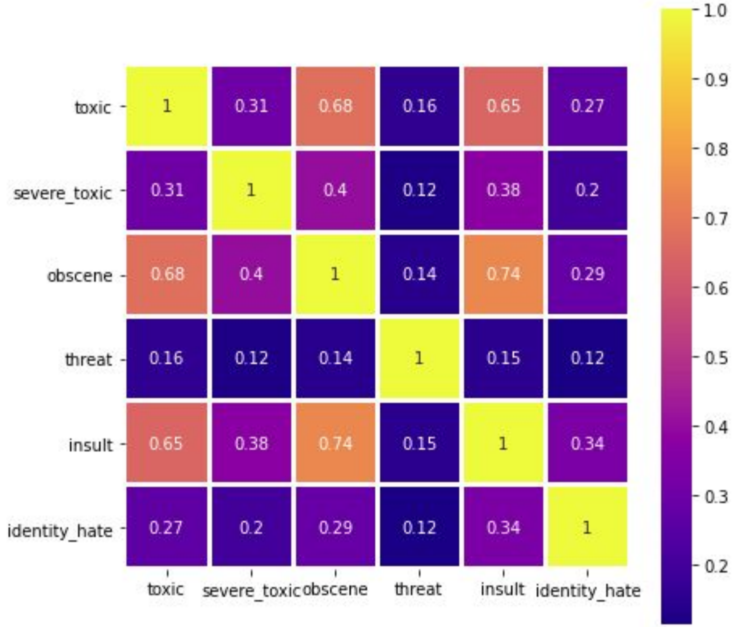
- ID
- Comment_text
- Toxicity indicator (toxic/ severe_toxic/ obscene/ threat/ insult/ identity_hate) - Categorical Data

Descriptive Analysis

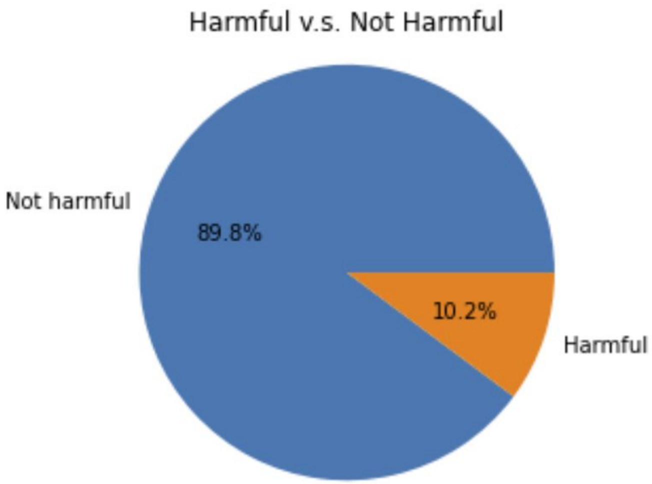




Combination	Count
Toxic	5666
Toxic + Obscene + Insult	3800
Toxic + Obscene	1758
toxic + Insult	1215
Toxic + Severe Toxic + Obscene + Insult	989
Toxic + Obscene + Insult + Identity Hate	618
...	...
Obscene + Threat + Insult	2
Obscene + Threat	2
Toxic + Severe Toxic + Threat + Identity Hate	1
Toxic + Severe Toxic + Threat + Insult	1



Pie Chart Visualization

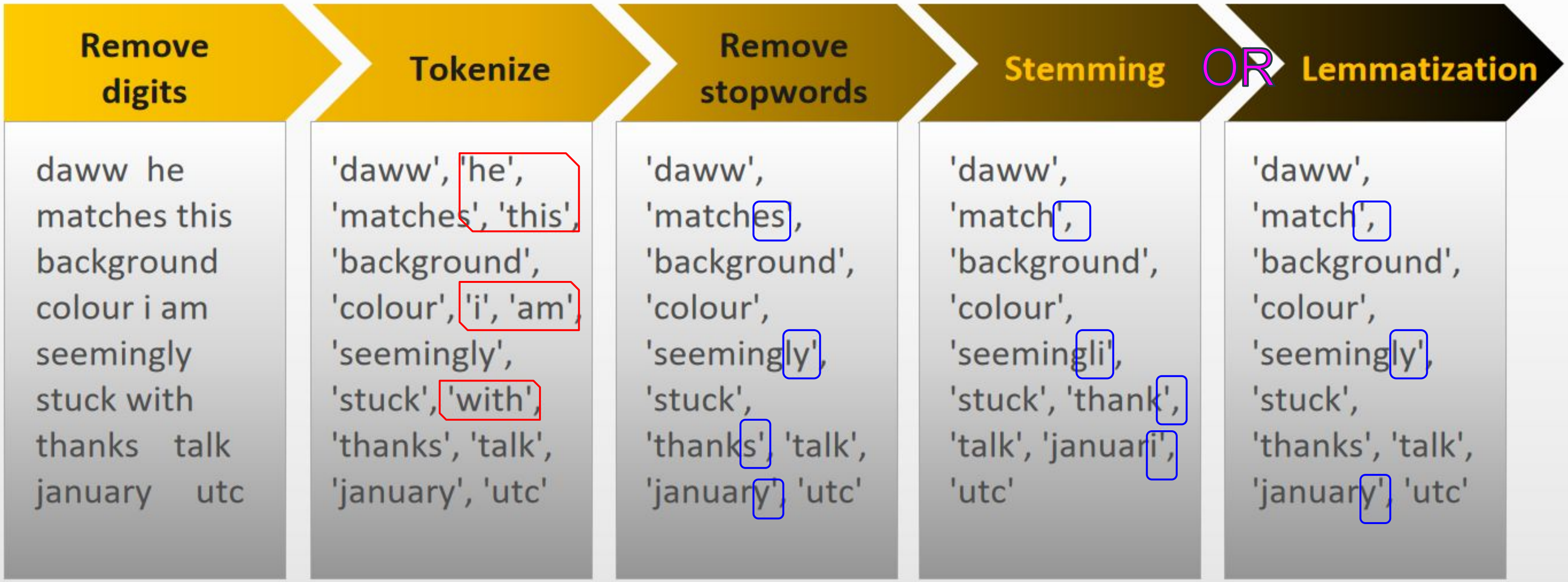


For the whole dataset:

Comment that are not harmful captures almost 90% of the data. Comments that are harmful capture 10.2% of the data.

Preprocess

Original Text	Expand Contractions	Lower Case	Remove punctuations	Remove digits
D'aww! He matches this background colour I'm seemingly stuck with. Thanks. (talk) 21:51, January 11, 2016 (UTC)	Daww! He matches this background colour I am seemingly stuck with. Thanks. (talk) 21:51, January 11, 2016 (UTC)	daww! he matches this background colour i am seemingly stuck with. thanks. (talk) 21:51, january 11, 2016 (utc)	daww he matches this background colour i am seemingly stuck with thanks talk 21 51 january 11 2016 utc	daww he matches this background colour i am seemingly stuck with thanks talk january utc


Remove digits
Tokenize
Remove stopwords
Stemming
OR
Lemmatization
daww he matches this background colour i am seemingly stuck with thanks talk january utc
'daww', 'he', 'matches', 'this', 'background', 'colour', 'i', 'am', 'seemingly', 'stuck', 'with', 'thanks', 'talk', 'january', 'utc'
'daww', 'matches', 'background', 'colour', 'seemingly', 'stuck', 'thanks', 'talk', 'january', 'utc'
'daww', 'match', 'background', 'colour', 'seemingli', 'stuck', 'thank', 'talk', 'januari', 'utc'
'daww', 'match', 'background', 'colour', 'seemingly', 'stuck', 'thanks', 'talk', 'january', 'utc'

Change of Comment after Preprocessing Steps

Step	Word changed/removed	Words unchanged/not removed	Percentage of words changed/removed	Percentage of words unchanged/removed
Stopwords Removal	1,135,347	1,217,470	48.25%	51.75%
Stemming	733,147	484,323	60.22%	39.78%
Lemmatization	399,716	825,840	32.62%	67.38%

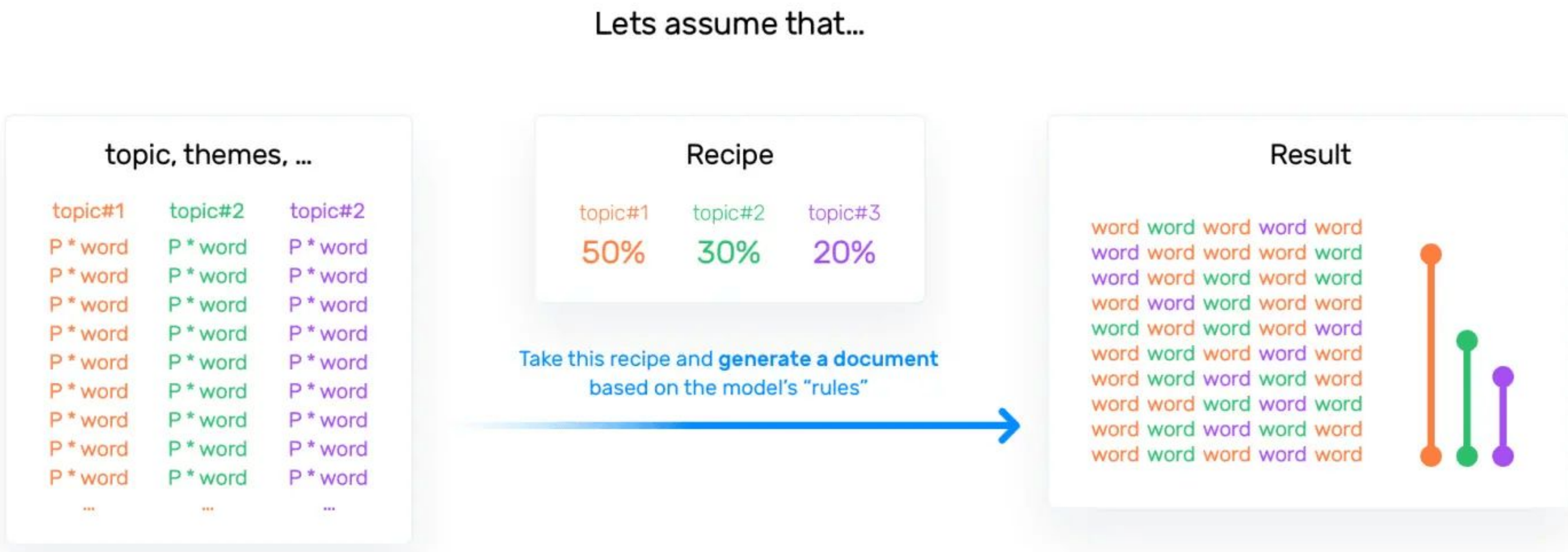
Outlier Detection

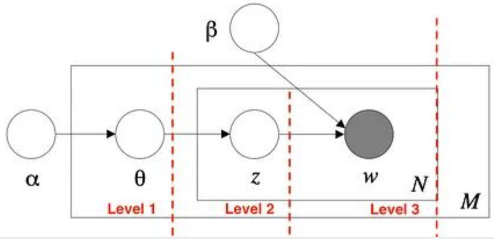
Text outlier detection is challenging because:

1. How to distinguish whether it is outlier or the natural variation/pattern in human language
2. Data Sparseness
3. Number of sub-groups
4. Distance concentration
5. …

We will apply outlier detection for our text data if necessary for if needed for analysis in future steps.

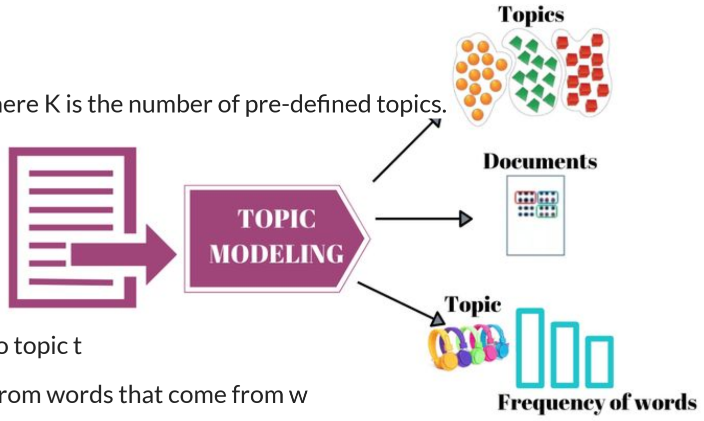
How does the LDA algorithm work?





How does the LDA algorithm work?

A Three-level hierarchical Bayesian model



1) For each document, randomly initialize each word to a topic amongst the K topics where K is the number of pre-defined topics.

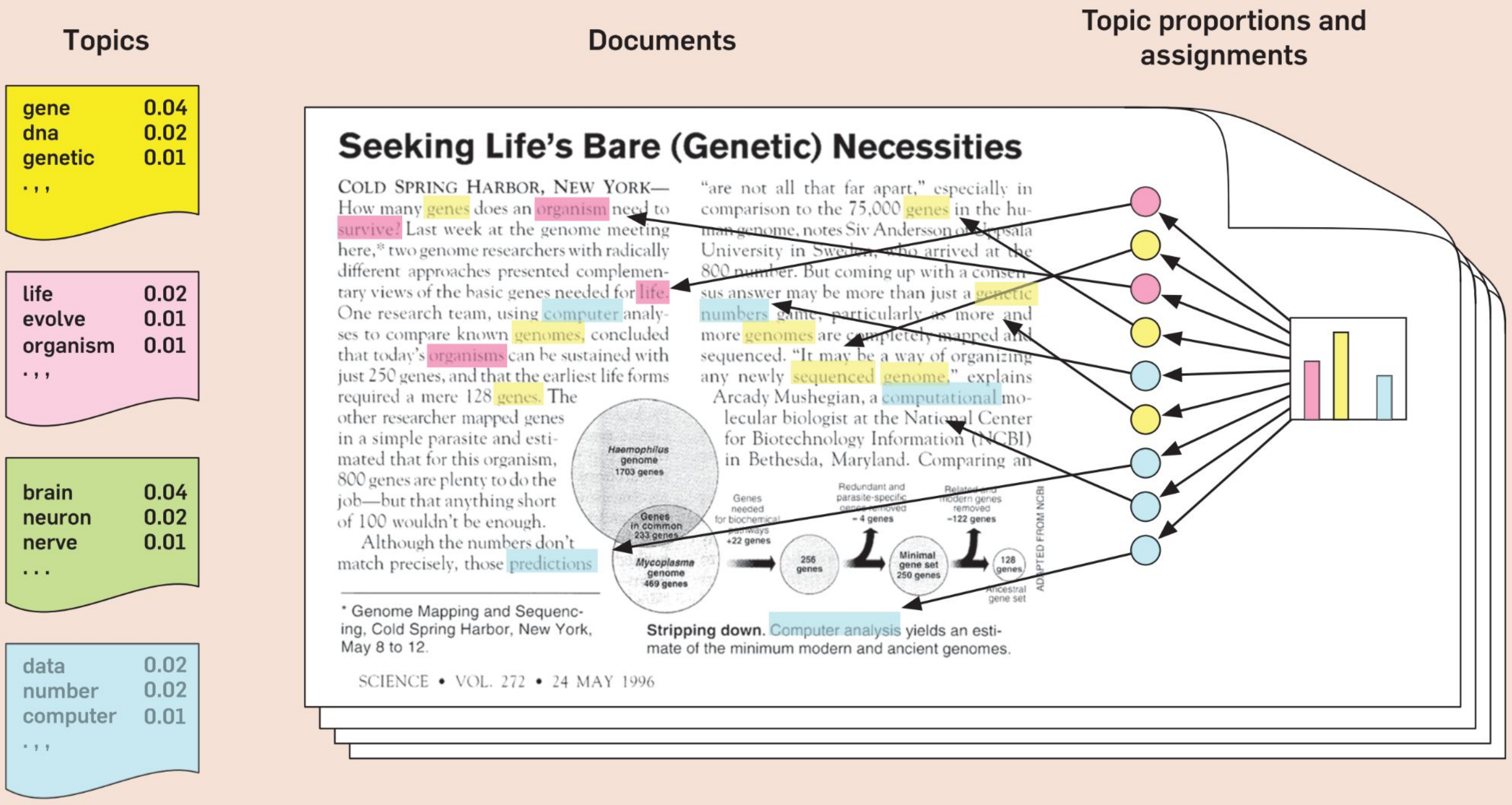
2) For each document d:

For each word w in the document, compute:

- P(topic t| document d): Proportion of words in document d that are assigned to topic t
- P(word w| topic t): Proportion of assignments to topic t across all documents from words that come from w

3) Reassign topic T’ to word w with probability p(t’|d)*p(w|t’) considering all other words and their topic assignments

The last step is repeated multiple times till we reach a steady state where the topic assignments do not change further. The proportion of topics for each document is then determined from these topic assignments.



Topics
Documents
Topic proportions and assignments
gene 0.04
dna 0.02
genetic 0.01
. , ,
life 0.02
evolve 0.01
organism 0.01
. , ,
brain 0.04
neuron 0.02
nerve 0.01
. . .
data 0.02
number 0.02
computer 0.01
. , ,
Seeking Life's Bare (Genetic) Necessities
COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.
Although the numbers don't match precisely, those predictions
"are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an
Haemophilus genome 1703 genes
Genes in common 233 genes
Mycoplasma genome 469 genes
Genes needed for biochemical pathways +22 genes
256 genes
Redundant and parasite-specific genes removed −4 genes
Minimal gene set 250 genes
Related and modern genes removed −122 genes
128 genes
Ancestral gene set
ADAPTED FROM NCBI
* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.
SCIENCE • VOL. 272 • 24 MAY 1996

Limitations of LDA

The static nature does not show the evolution of topics over time

Inability in capturing correlations

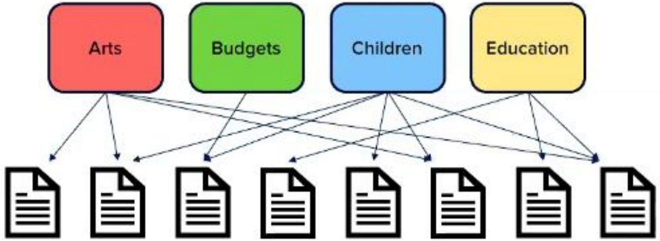
Algorithm’s inherent instability

Simplifying “bag-of-words” exchangeability assumption

Necessity of a fixed *k* value

Doesn't define the topic on its own

Understanding Topic



Topic: 0

Words: ['eat', 'sex', 'idea', 'notrhbysouthbanof', 'famili', 'info', *******, 'fascist', 'hide', '****', 'activ', 'share', 'coward', 'ladi', 'ear', 'hot', 'choic', 'here', 'wife', 'da', 'victim', 'bot', 'mod', 'al', 'bite', 'yea', 'knock', 'agent', 'shithol', 'handl']

Topic: 1

Words: ['aid', 'fat', '***', ******, '***', ******, 'million', 'bullshit', 'f', 'k', 'white', 'report', 'continu', 'c', 'pictur', 'dad', 'g', 'explain', 'dumb', 'school', 'bigot', 'fight', 'bit', 'hitler', 'respons', 'wish', 'term', 'final', 'disrupt', 'suggest']

Topic: 2

Words: ['hate', 'hi', 'made', 'discus', 'claim', 'ha', 'without', 'alreadi', 'death', 'happen', 'yet', 'refer', 'accus', 'complet', 'le', 'english', 'utc', 'includ', 'remark', 'list', 'watch', 'cite', 'controversi', 'excus', 'pov', 'john', 'dare', 'rude', 'thread', 'commun']

Topic: 3

Words: ['page', 'get', 'shit', 'know', 'edit', 'freedom', 'peopl', 'hey', 'articl', 'admin', 'block', 'talk', 'stop', 'one', 'cocksuck', 'delet', 'think', 'would', 'vandal', 'plea', 'say', 'keep', 'bad', 'tri', 'make', 'dont', 'idiot', 'right', 'see', 'comment']

...

How to understand each topic

- ngrams for each category using TF*IDF
- Generate a list from Wikipedia titles, extract keyphrases, predict the related wikipedia pages and use the keyphrases.
- Generate a hand-labeled dataset.
- Use a graph populated with topics and the relations between words and topics to predict the most likely topics
- Abstractive summarization and keyphrase extraction

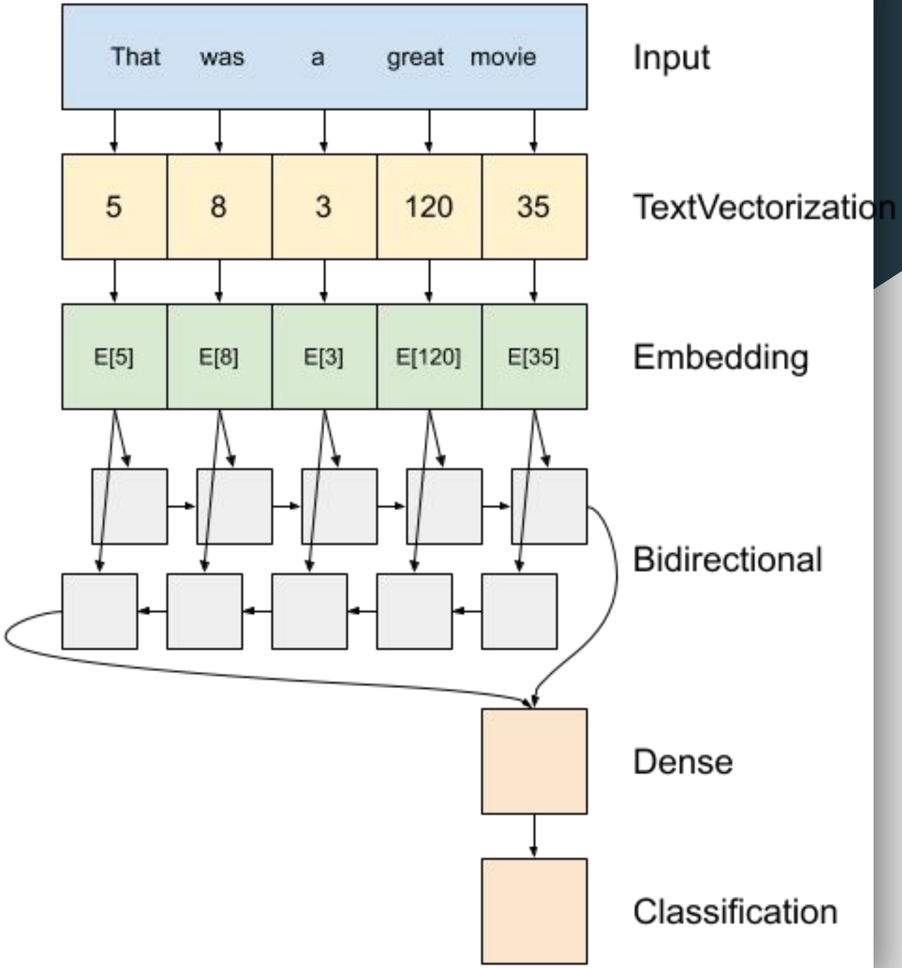
LSTM

LSTM: long short-term memory networks

LSTM vs RNN:

- Long-term dependence
- Vanishing gradients
- Forget gate

More efficient !



Machine Learning Morphism

$$\mathcal{ML} = \mathcal{ML}9 \circ \mathcal{ML}8 \circ \mathcal{ML}0\text{-}7 =$$

Input Space: X^n text data, n = 155k observations

Output Space: [0, 1]

Learning Morphism: $F(\boldsymbol{x};\ \theta_{0\text{-}7}, \boldsymbol{w}, \theta_9) = F_9 \circ F_8 \circ F_{0\text{-}7}$

Parameter Prior: $P(\boldsymbol{x};\ \theta_{0\text{-}7}, \boldsymbol{w}, \theta_9) = 1$

Empirical Risk Function: $\sum 1(F(x_i;\ \theta_{0\text{-}7}, \boldsymbol{w}, \theta_9) = y_i)/$# of test data

$\mathcal{ML}0\text{-}7$: Data Preprocessing: Contractions Expansion, ..., Lemmatization

$\mathcal{ML}8$: Long Short Term Memory

$\mathcal{ML}9$: Evaluation for accuracy

Next Steps

Week	Agenda
9	Cultivate LSTM network; Hypermeter tuning; Adjust preprocessing
10-11	Explore More models; Figure abnormal sample; Optimization; Add more visualization
12	Final check; Catch up progress; Prepare presentation
13	Presentation

Reference

https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge/data
https://builtin.com/machine-learning/nlp-machine-learning
https://analyticsindiamag.com/pseudo-labelling-a-guide-to-semi-supervised-learning/
Text Messages Classification using LSTM, Bi-LSTM, and GRU | by Nuzulul Khairu Nissa | MLearning.ai | Medium
An Improved LSTM Structure for Natural Language Processing
BERT and fastText Embeddings for Automatic Detection of Toxic Speech
https://en.wikipedia.org/wiki/Vanishing_gradient_problem