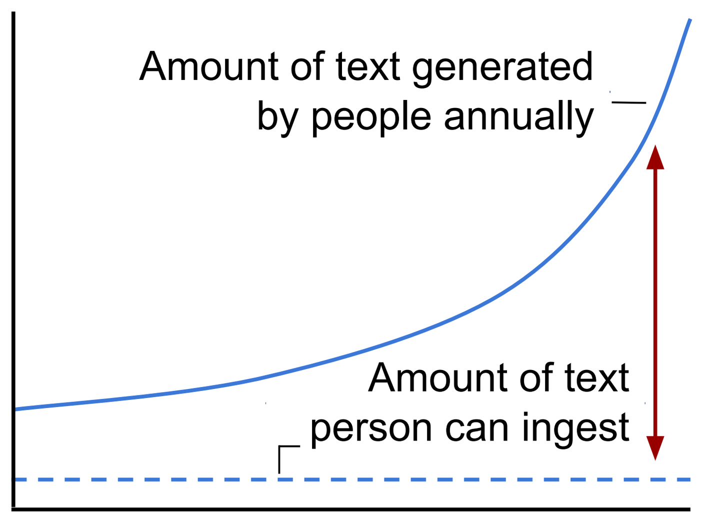
Paragraph Topic Extraction

- From Naive Bayes to Convolutional Neural Network -

Edward Ng, Eugene Nho
Stanford University Dept. of Electrical Engineering, Stanford Graduate School of Business
{edjng, enho}@stanford.edu

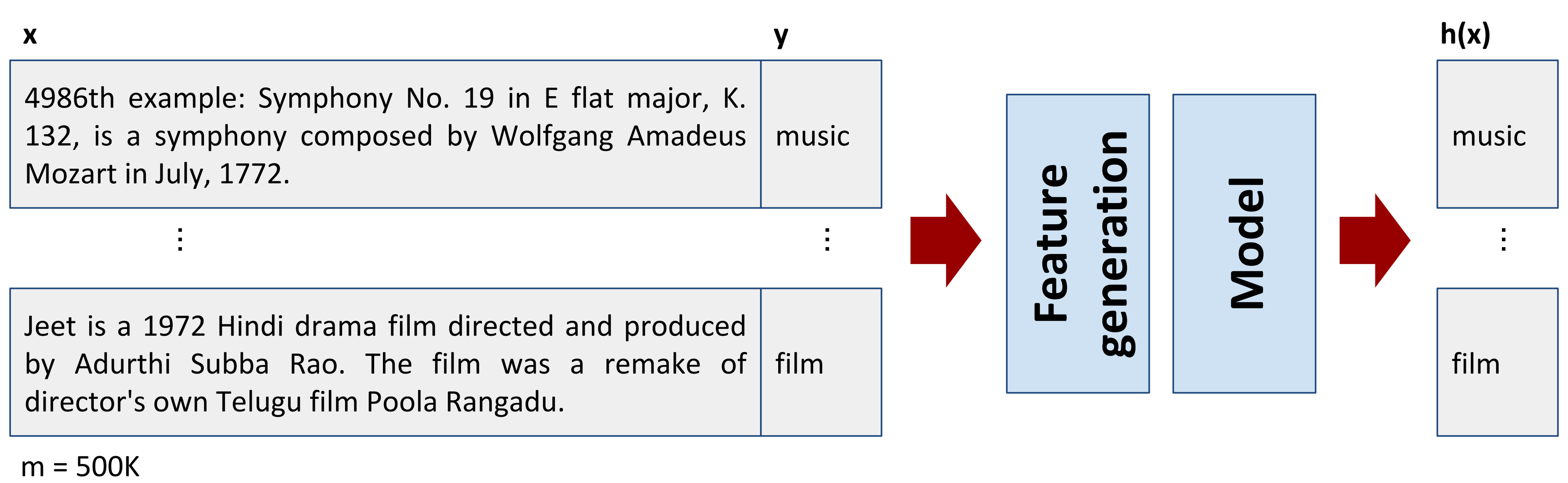
Motivation & Problem Definition



- "Exponentially" increasing information vs constant human ability to ingest text. Automatic topic extraction can help close this gap
- Example use cases: detecting relevant sections from SEC filings, market reports, legal documents, etc.

Formalized Problem: Given a paragraph of text as input, predict what topics the input is about, from a given set of topics (multilabel)

Overall Approach: Schematic



Overall Approach: Data, Features, and Model

Data

- Wikipedia "abstract" (first paragraph) as input (x), their category assignments as labels (y)
- Total 500K inputs, labeled as one or more of {math, politics, computer science, film, music} (multilabel)
- Wikipedia chosen because of readily available human-tagged labels

Features

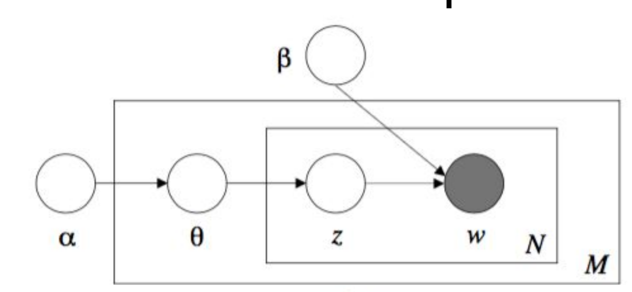
- Term Frequency-Inverse Document Frequency (tf-idf): Common text normalization technique
- GloVe: Richer representation; captures co-occurrence of words, which helps topic detection

Model

We focused on first building a "quick-and-dirty" baseline, then added more sophisticated features / models and observed how performance improved[1]:

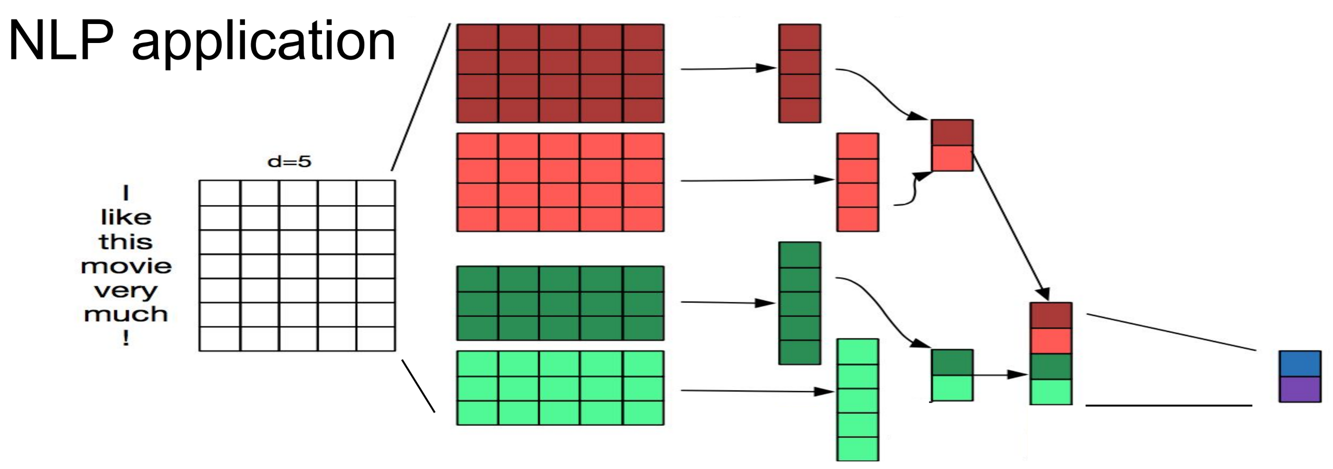
Model [features in bracket]	**Rationale**
1 Naive Bayes [tf-idf]	Common baseline model for text classification
2 One-vs-rest classifier (OvR) [GloVe]	One-vs-rest supports multilabel learning; richer feature (GloVe)
3 Latent Dirichlet Allocation (LDA) + OvR [tf]	To capture latent topics more effectively
4 LDA + GloVe + OvR [tf]	Complementary: GloVe (local focus) and LDA (more global)
5 Convolutional Neural Network (CNN) [GloVe]	Fast; generalizes well; local word order is not important

4 LDA: document - topic - word representation



$$p(\mathbf{w}|\alpha,\beta) = \int p(\theta|\alpha)\left(\prod_{n=1}^{N}\sum_{z_n} p(z_n|\theta)p(w_n|z_n,\beta)\right)d\theta.$$

5 CNN: NLP application



Results and Discussion

F1 score of various approaches

	NB	OvR + GloVe	LDA	LDA + GloVe + OvR	CNN
Precision		76	62	82	92
Recall		48	78	66	84

- As we utilized richer representations of input text, performance (F1 score) increased overall
- Power of latent topic representation (LDA) is notable, with ~1.6x increase in F1 score from NB (vs ~1.25x GloVe)
- Topics captured by LDA closely mirror original labels, e.g. "Topic 3" represented by words {album, song, released, single, ...}, "Topic 4" by {theory, displaystyle, soviet, university, graph analysis, mathematical, ...}
- CNN provided superior results despite its primary use case being image classification
- Future work: (1) experimenting with different hyperparameter settings for CNN
  (2) implementing RNN (more "classic" go-to architecture for NLP) and comparing with CNN

1 `sklearn`, `numpy`, `tensorflow`, and `keras` were used in our development. Our classifiers assumed the default hyperparameters provided by the respective libraries
2 LDA diagram: D. M. Blei, A.Y. Ng, and M.I. Jordan. (2003). Latent Dirichlet allocation. *JMLR*.
3 CNN diagram: Zhang, Y., & Wallace, B. (2015). A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification