

Applied Text Mining in Python

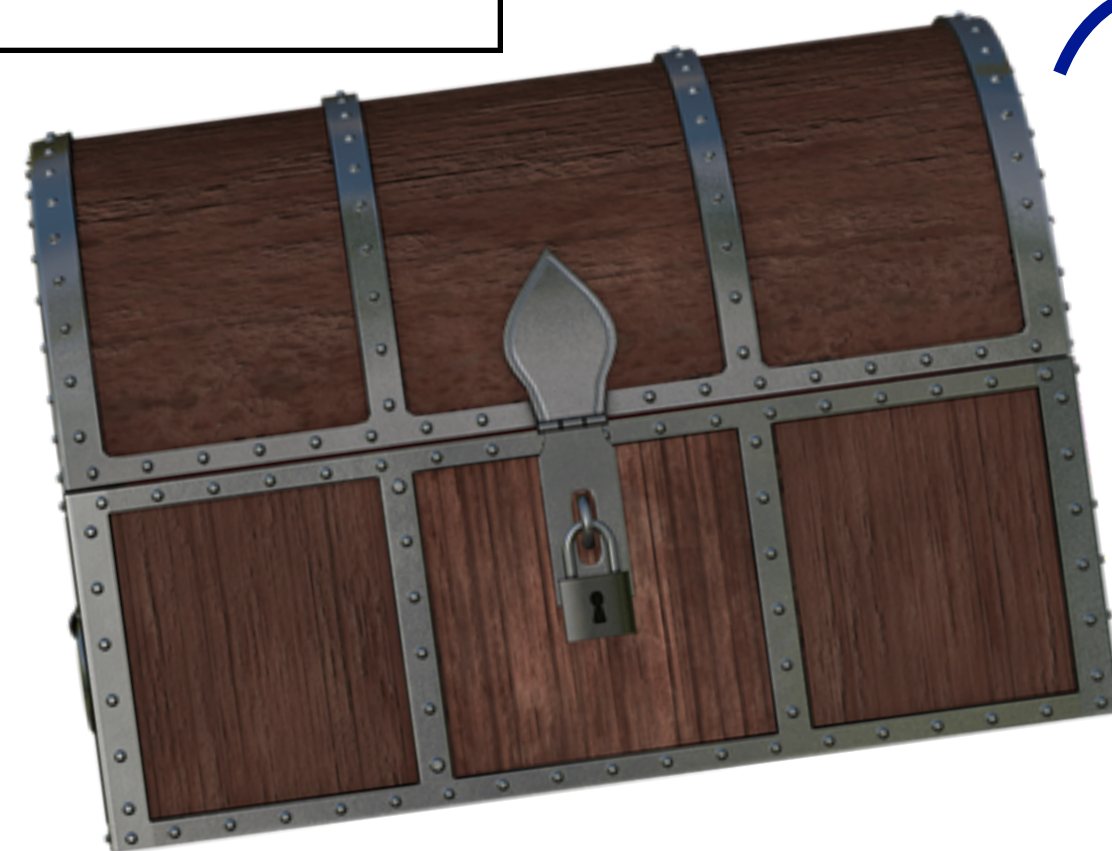
Generative models and LDA

a probability
distribution of
how likely it is to
see the words

Generative Models for Text

$\text{Pr}(\text{text} \mid \text{model})$

the 0.1
is 0.07
harry 0.05
potter 0.04
movie 0.04
plot 0.02
time 0.01
rowling 0.01

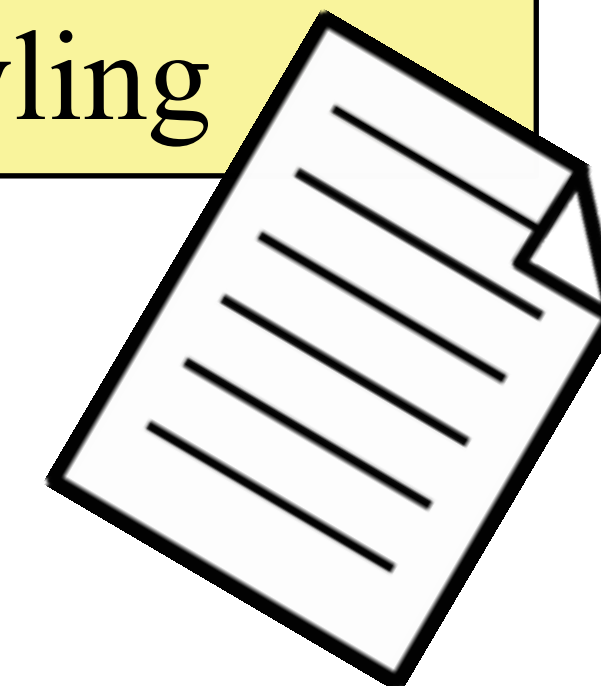


Inference, Estimation



the
harry
potter
movie
is
harry
is

the ... movie ...
harry potter ...
is ... based on ...
j. k. rowling



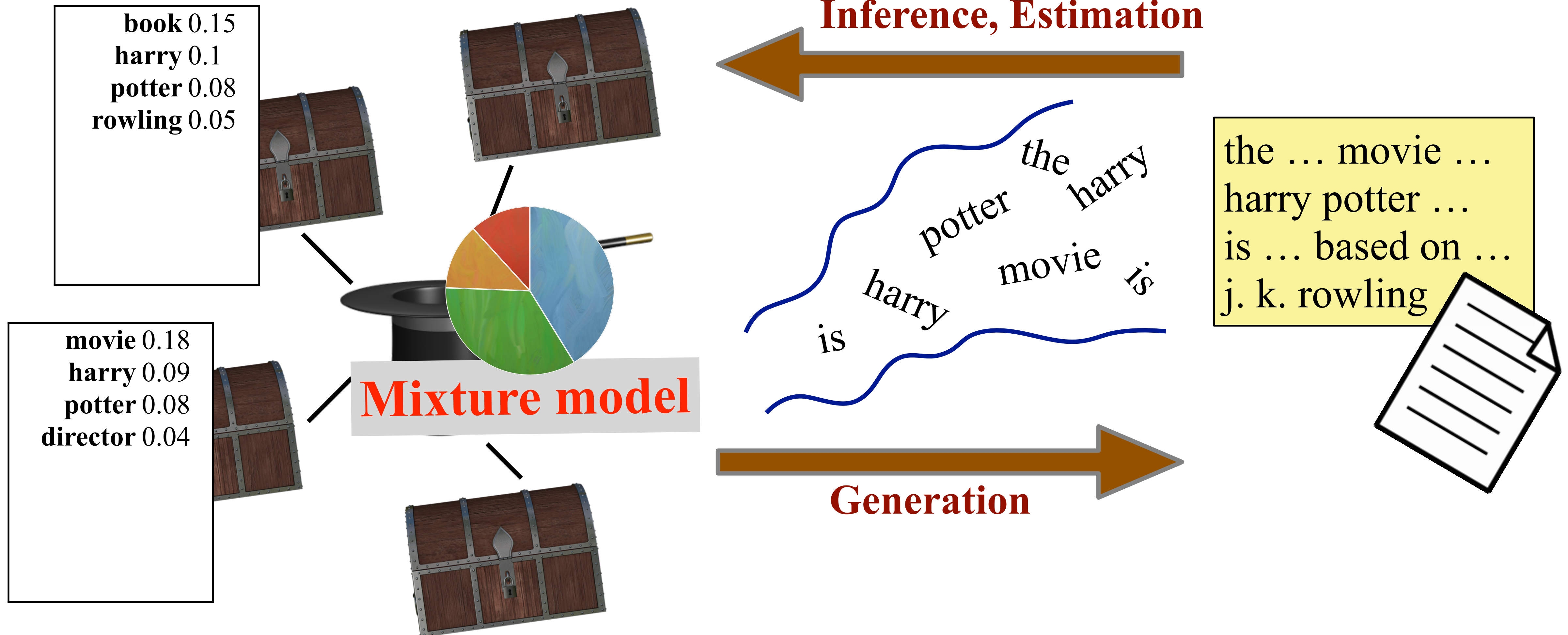
Generation



one document generated from 4 topics

Generative Models can be Complex

$\text{Pr}(\text{text} \mid \text{model})$



Latent Dirichlet Allocation (LDA)

- Generative model for a document d
 - Choose length of document d
 - Choose a mixture of topics for document d
 - Use a topic's multinomial distribution to output words to fill that topic's quota

Topic Modeling in Practice

- How many topics?
 - Finding or even guessing the number of topics is hard
- Interpreting topics
 - Topics are just word distributions
 - Making sense of words / generating labels is subjective

Topic Modeling: Summary

- **Great tool for exploratory text analysis**
 - **What are the documents (tweets, reviews, news articles) about?**
- **Many tools available to do it effortlessly in Python**

Working with LDA in Python

- Many packages available, such as gensim, lda
- Pre-processing text
 - Tokenize, normalize (lowercase)
 - Stop word removal
 - Stemming
- Convert tokenized documents to a document - term matrix
- Build LDA models on the doc-term matrix

Working with LDA in Python (2)

- **doc_set: set of pre-processed text documents**

```
import gensim
```

```
from gensim import corpora, models
```

```
dictionary = corpora.Dictionary(doc_set) Create the dictionary, a mapping between IDs and words
```

```
corpus = [dictionary.doc2bow(doc) for doc in doc_set] bag of words, document-term matrix
```

```
ldamodel = gensim.models.ldamodel.LdaModel (corpus, num_topics=4,  
id2word=dictionary, passes=50) number of topics you want to learn, mapping, how many passes you like
```

```
print(ldamodel.print_topics(num_topics=4, num_words=5))
```

tope 5 words for the topics

- **ldamodel can also be used to find topic distribution of documents**

```
topic_dis = ldamodel[new_doc]
```


Take Home Concepts

- **Topic modeling is an exploratory tool frequently used for text mining**
- **Latent Dirichlet Allocation is a generative model used extensively for modeling large text corpora**
- **LDA can also be used as a feature selection technique for text classification and other tasks**