Clustering Television Shows Using Text

Problem Statement

I used transcripts of television episodes to look for patterns between shows using the Latent Dirichlet Allocation (LDA) model, which clusters similar shows based upon the two shows use similar words at similar frequencies.

About the Data

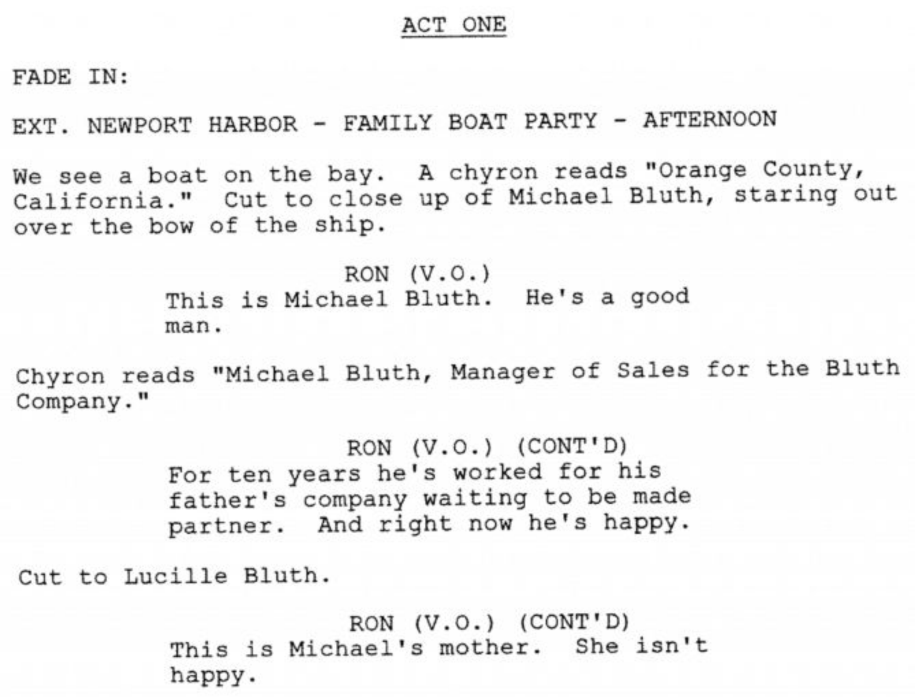
Source: springfieldspringfield.co.uk

- 117,937 television transcripts
- 4,667 television shows

Including every episode of Doctor Who (50 years of TV!)

To 1600 Penn (1 season from late 2000's)

Note: Script vs Transcript



ACT ONE

FADE IN:

EXT. NEWPORT HARBOR - FAMILY BOAT PARTY - AFTERNOON

We see a boat on the bay. A chyron reads "Orange County, California." Cut to close up of Michael Bluth, staring out over the bow of the ship.

RON (V.O.)
This is Michael Bluth. He's a good man.

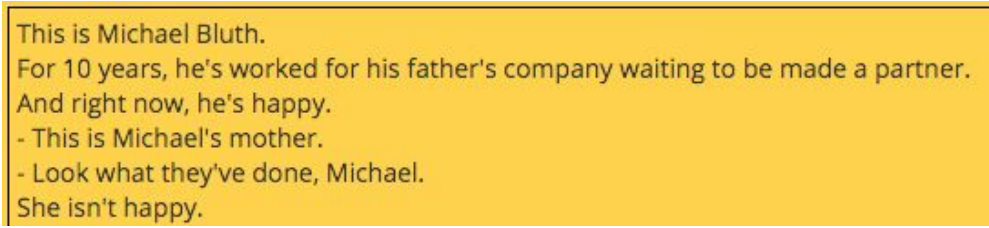
Chyron reads "Michael Bluth, Manager of Sales for the Bluth Company."

RON (V.O.) (CONT'D)
For ten years he's worked for his father's company waiting to be made partner. And right now he's happy.

Cut to Lucille Bluth.

RON (V.O.) (CONT'D)
This is Michael's mother. She isn't happy.

Original Script



This is Michael Bluth.
For 10 years, he's worked for his father's company waiting to be made a partner.
And right now, he's happy.
- This is Michael's mother.
- Look what they've done, Michael.
She isn't happy.

My Data

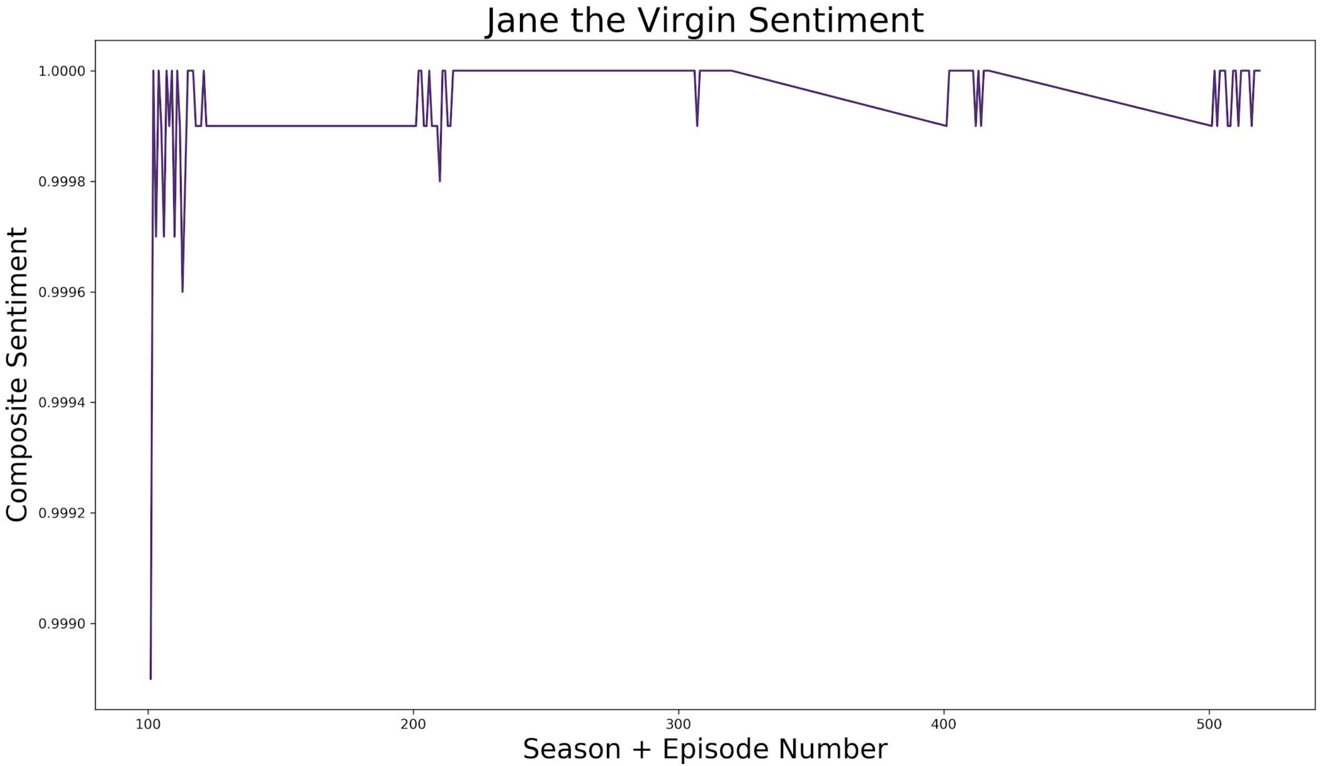
Sentiment Analysis of Episodes

The text of each episode was run through nltk's VADER sentiment analysis to calculate the composite sentiment.

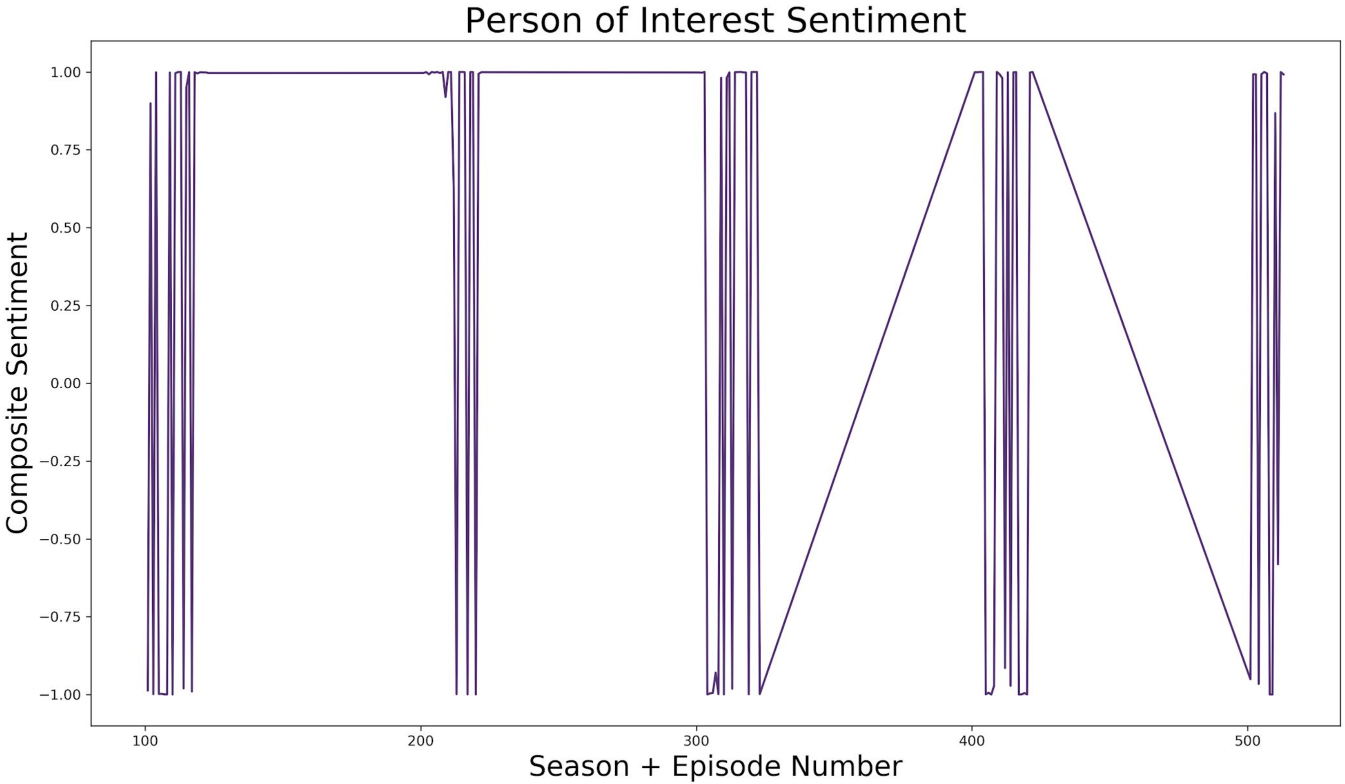
98.86% of all episodes have a sentiment score either greater than .75 or less than -.75.

Further Research: it seems that comedies have consistently positive sentiment scores, save for the occasional negative episode, while dramas seem to have an oscillating sentiment score

Comedy


Jane the Virgin Sentiment
Composite Sentiment
1.0000
0.9998
0.9996
0.9994
0.9992
0.9990
100
200
300
400
500
Season + Episode Number

Drama


Person of Interest Sentiment
Composite Sentiment
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
100
200
300
400
500
Season + Episode Number

Cleaning the Data

The column containing the television show name was irregularly formatted, it either looked like:

Arrested Development s01e01 Episode Script

OR

Andy Richter Controls the Universe (2002) Episode Scripts

Also removed escape and other characters

Tokenizing/Stop Word Removal/Lemmatizing

Tokenizing: break the text down into individual words

Stop Words: Words that either occur too frequently to be useful or are too strongly associated with a topic or show

Stop words list was created from a combination of 4 online sources and 1 list I made myself

Final count: 2,554 stop words

Lemmatizing: breaking a word down to its base: turning “ran” into “run,” “dogs” to “dog”

The Model: Latent Dirichlet Allocation (LDA)

LDA is an unsupervised clustering algorithm that measures the frequency of words and other similar words and groups documents into different clusters based upon their probability of belonging to that group (like a softmax) and also measures how similar a document is to all of the other documents in the corpus.

Tuning the Model

The one hyperparameter in LDA is the number of clusters: I ran the model for 23 different values of k and compared their coherence scores (a range from 0 to 1 that measures how disjoint clusters are), ranging from 5 to 500. The coherence scores were relatively stable but k=10 had the highest value at .2749 and became my final model

Results and Topic Clusters

The best LDA model had 10 clusters with a coherence score of .2749. LDA is unsupervised so any meaning found in the clusters is through interpretation.

2 Example Clusters:

cluster 6:
['money','school','work','police','doctor','murder','buy','lie','gun','drive']

cluster 2:
['money','lie','hate','school','fire','drive','fall','hang','doctor','death']

Issues

TOO MUCH TEXT

Model took a long time to run and could only be run on a virtual machine

Mallet LDA algorithm could not handle as much data as I had

Lemmatizing was major time issue

GENSIM DOCUMENTATION

The different parameters of the model are not clearly explained and how to access the attributes of the classes are not obvious

Next Steps

Get a recommendation system to work

Turn it into a Flask app

Sources

- https://www.springfieldspringfield.co.uk
- https://www.scriptreaderpro.com/best-tv-scripts/
- https://www.machinelearningplus.com/nlp/topic-modeling-gensim-python/
- http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf