Theme classification based on article titles

Building a system that facilitates manual multiple-choices among massive number of categories

Problem

- Article sharing platforms (blog/journal …) need indexing systems:
  - Key-word based:
    flexible, dynamic
    provided by author
  - Theme based:
    convenient, low-maintenance
    limited number of themes set by system

  How to help author choose the best theme?
  - Algorithm that chooses for author
  - Algorithm that recommend themes for the author to choose ✓

Application for businesses

- Similar problem for businesses: ticket-handling system

  Determine possible departments for incoming requests

- Options:
  1. Fully manual system – too many categories (classes) for operator to remember, expensive
  2. Fully automatic system – low precision
- Alternative:

  Semi-automatic recommendation system to suggest possible departments for the request – increase operator efficiency

Data source and cleaning

- 126,418 articles from Medium (medium.com)
  - classified in 93 categories by the platform
- Data cleaning:
  - remove punctuation marks
  - convert to lower case

CATEGORY	TITLE	TITLE (CLEANED)
WORK	"21 Conversations" - A fun (and easy) game for...	21 conversations a fun and easy game for...
SPIRITUALITY	"Biblical Porn" at Mars Hill	biblical porn at mars hill
LGBTQIA	"CISGENDER?! Is That A Disease?!"	cisgender is that a disease
EQUALITY	"Call me Nat Love" :Black Cowboys and the Fron...	call me nat love black cowboys and the fron...
ARTIFICIAL-INTELLIGENCE	"Can I Train my Model on Your Computer?"	can i train my model on your computer

Exploratory analysis and data visualization

Wordclouds

Cosine similarity and hierarchical clustering

Wordclouds
popular (tf-idf) words in example categories

Feature visualization: worldclouds are able to show important features (words) for each category.

- Upper-right: election-2020
- Upper-left: money
- Lower-left: travel
- Lower-right: sexuality

Cosine similarity

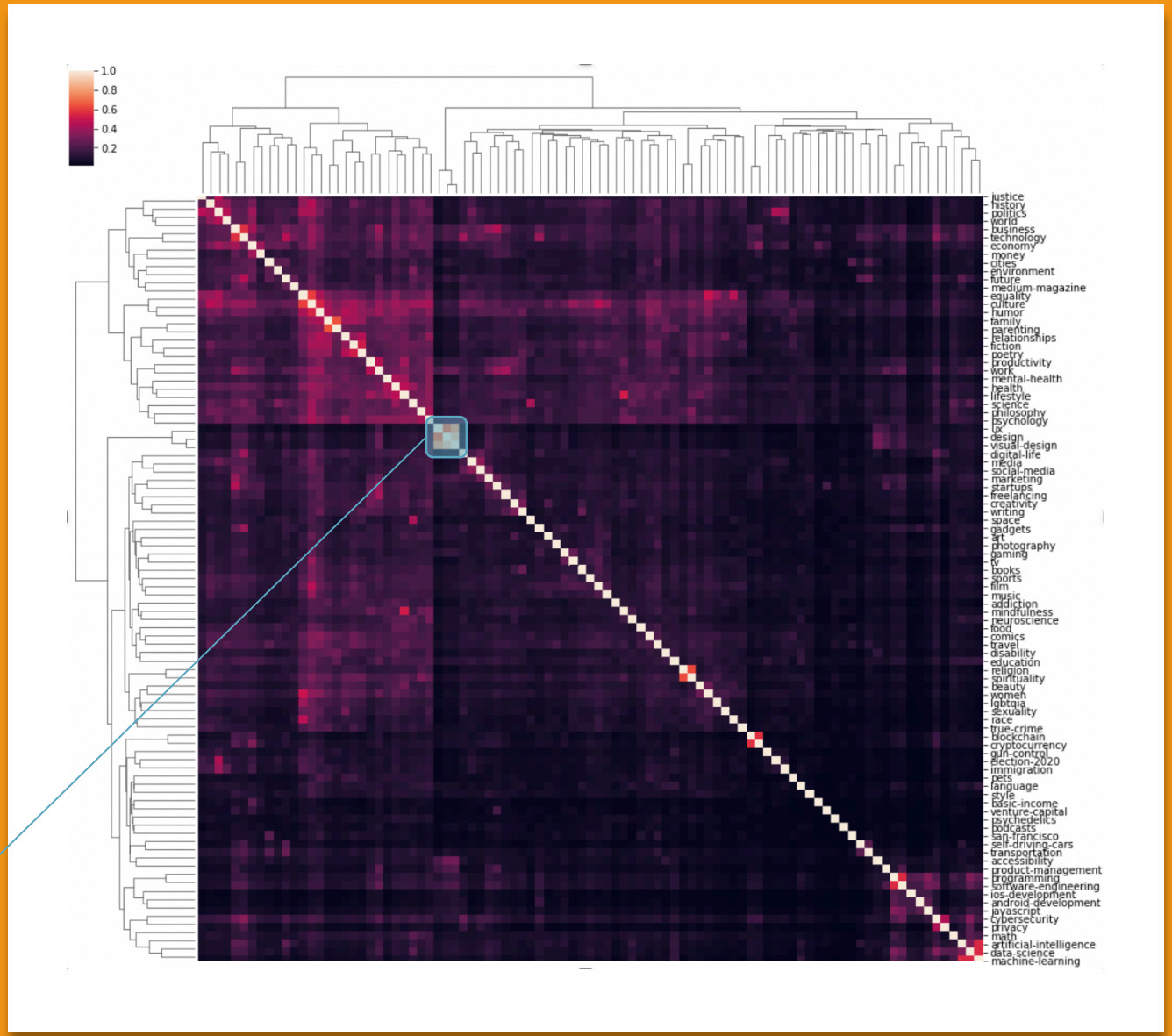
Bag-of-words model generates a vector representation for each title using TF-IDF

For each category, a mean vector can be computed by averaging all title vectors within the category.

Cosine similarity matrix of categorical average vector →

categories with intuitive relations grouped together by hierarchical clustering

ux, design, visual-design



Classifier models

- Benchmark:

Naïve Bayes

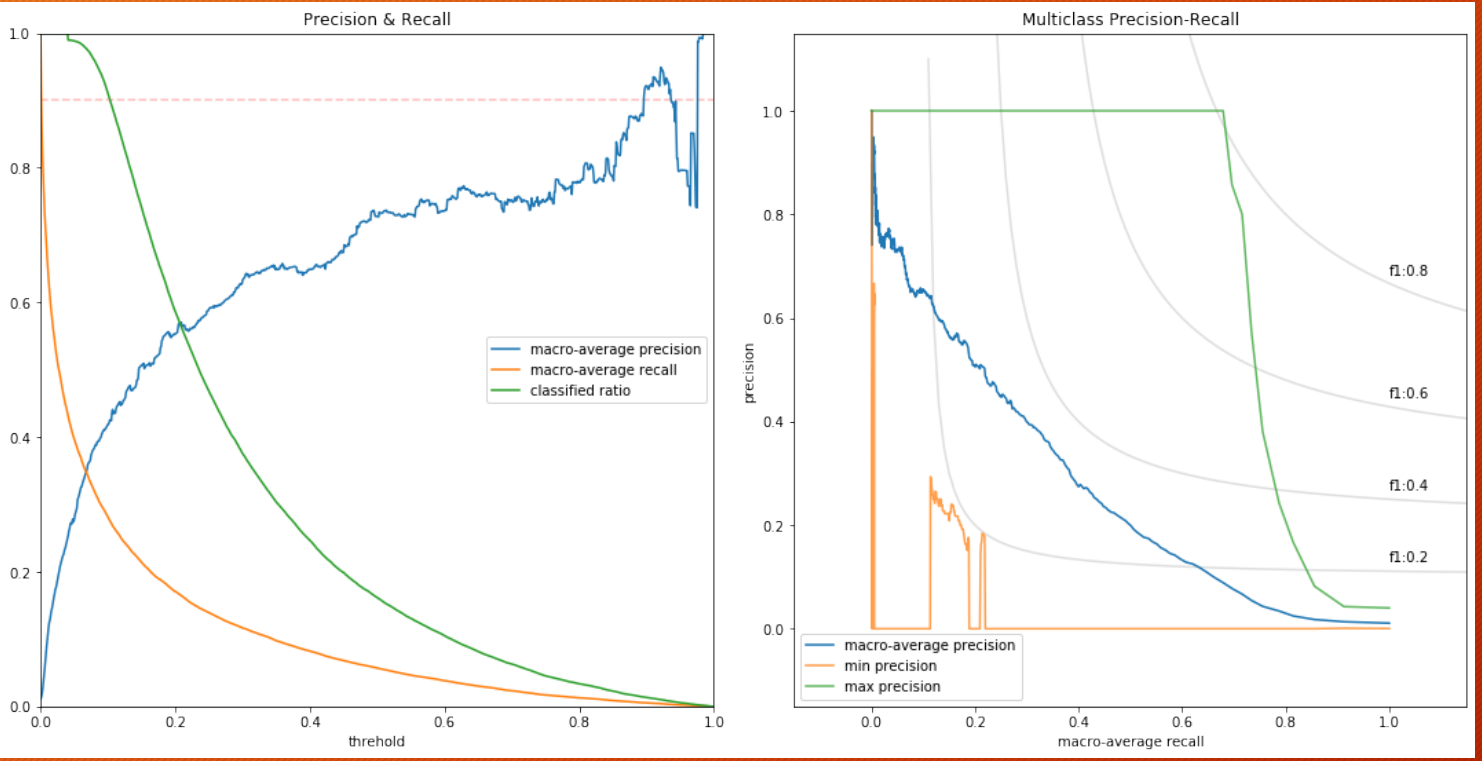
- Neural networks:

Multilayer perceptron network, recurrent network, long short-term memory network

Naïve Bayes classifier

- 20% test set, 80% training set
- TF-IDF vectorization
- Hyperparameter tuning – 5-fold cross-validation:
  - n-grams
  - threshold and cut-off frequency for a word to be included in the vocabulary
  - pseudo-count for additive smoothing
- accuracy on test set: 35%, average multiclass precision: 46%

Naïve Bayes classifier

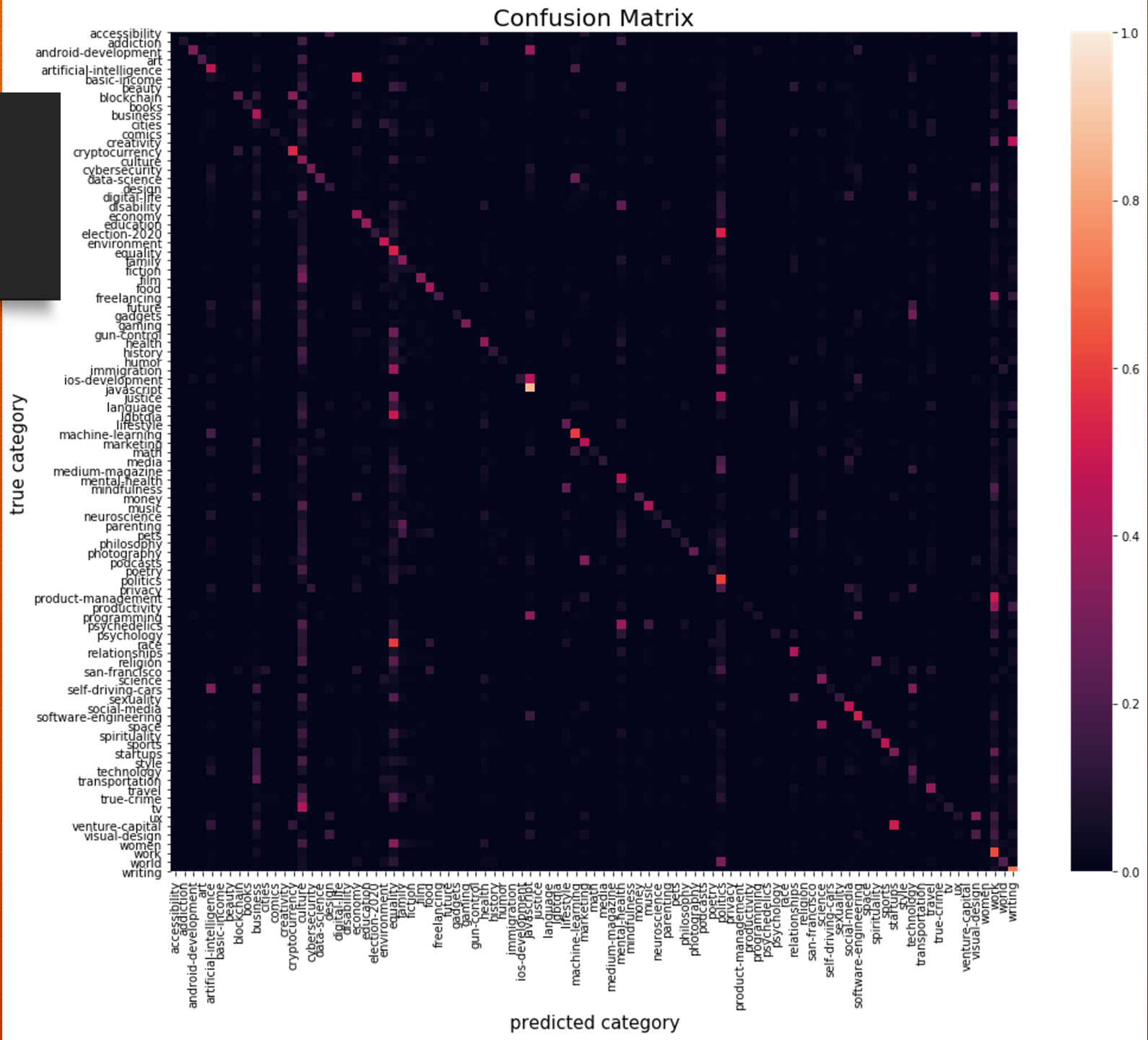


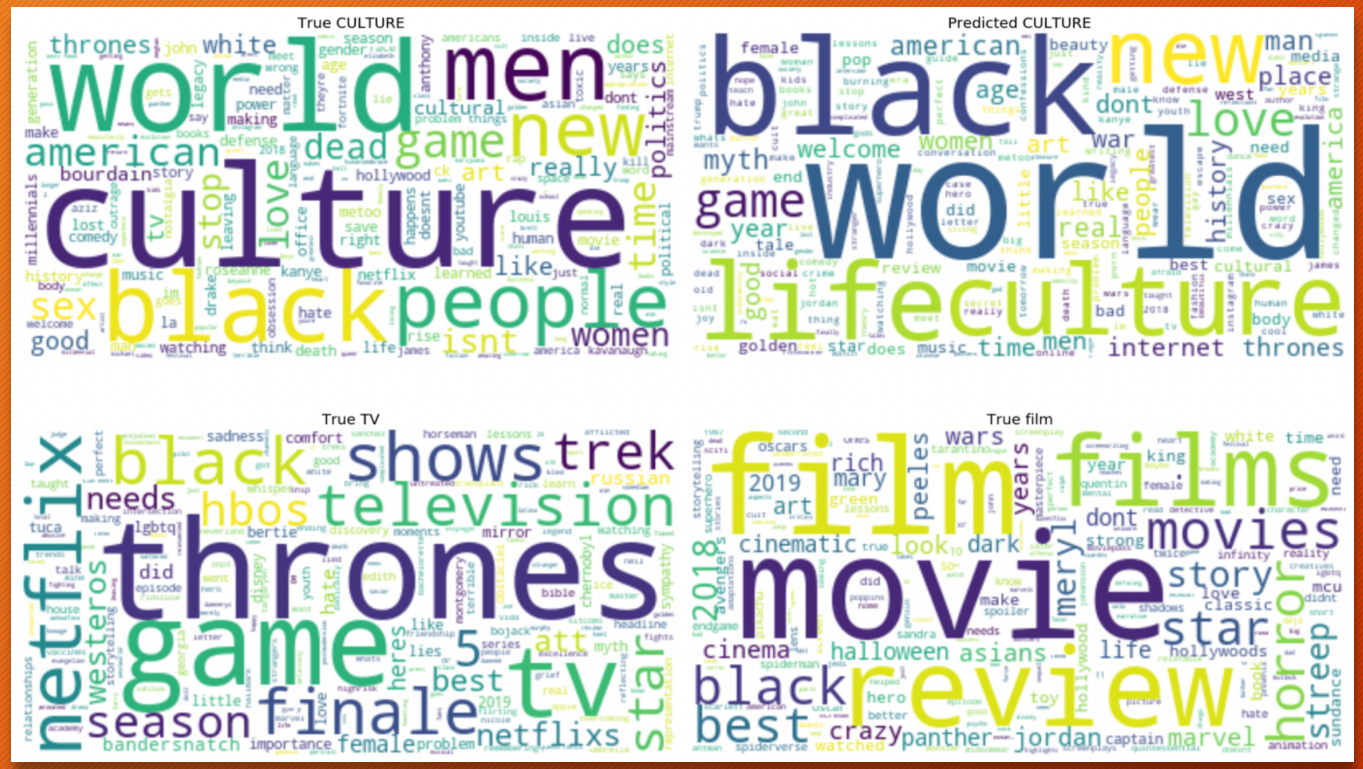
To achieve an average precision more than 90%, the decision threshold should be no less than about 0.90. At that level, the average recall is no more than 0.5%.

Naïve Bayes classifier

Confusion matrix

- Many categories misclassified into *culture*
- *Tv* and *film* caused major confusions





Tv and *film* were misclassified as *culture*.

- Both words 'movie' and 'tv' are included in the wordcloud of the true *culture*
- A lot of other major features are also shared by either *film* or *tv* with *culture* – 'hollywood', 'game', 'thrones', 'black', 'netflix'…

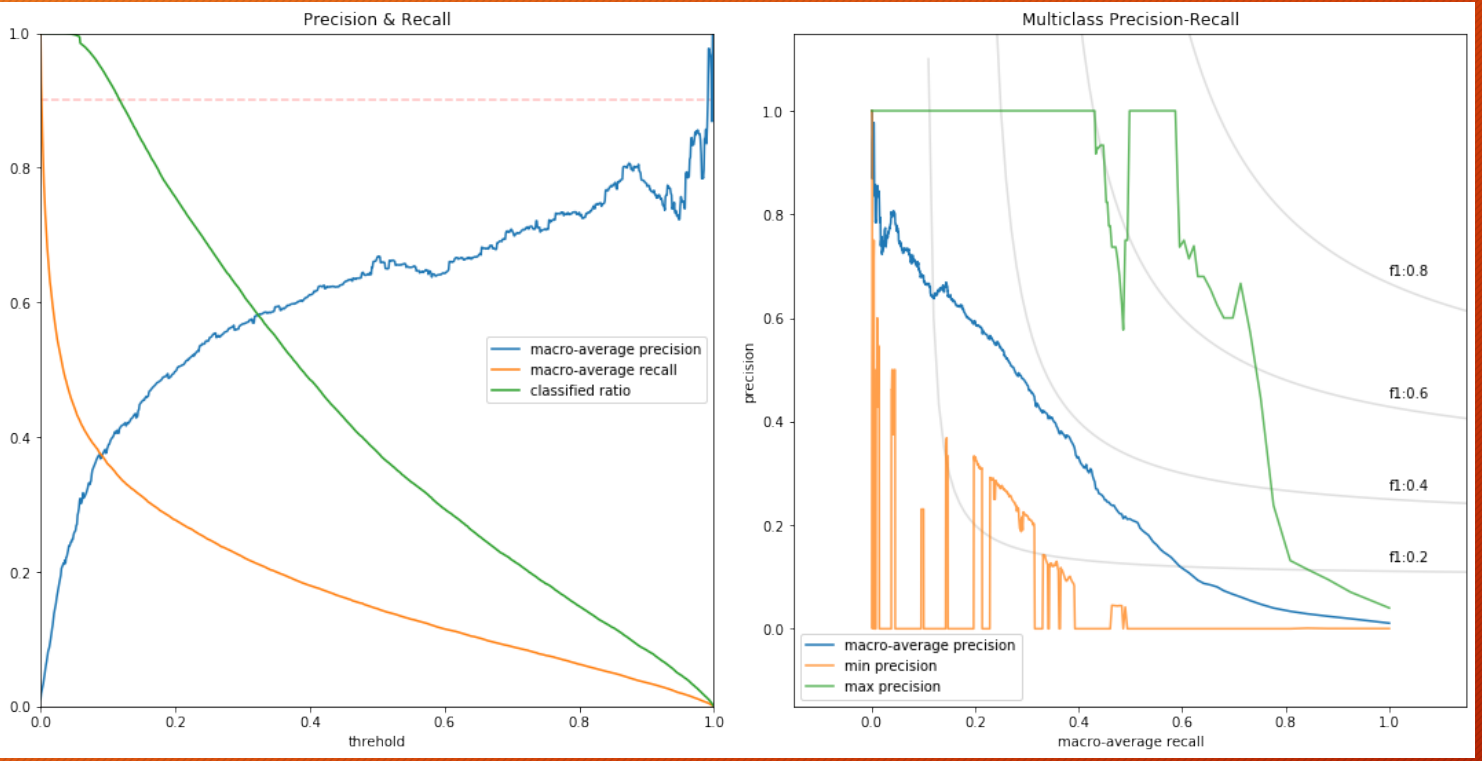
Naïve Bayes classifier

Wordclouds of *culture*, *tv* and *film*

Multilayer perceptron network

- 20% test set, 80% training set
- TF-IDF vectorization
- Architecture: 2 hidden layers of 256 units
- accuracy on test set: 41%, average multiclass precision: 45%

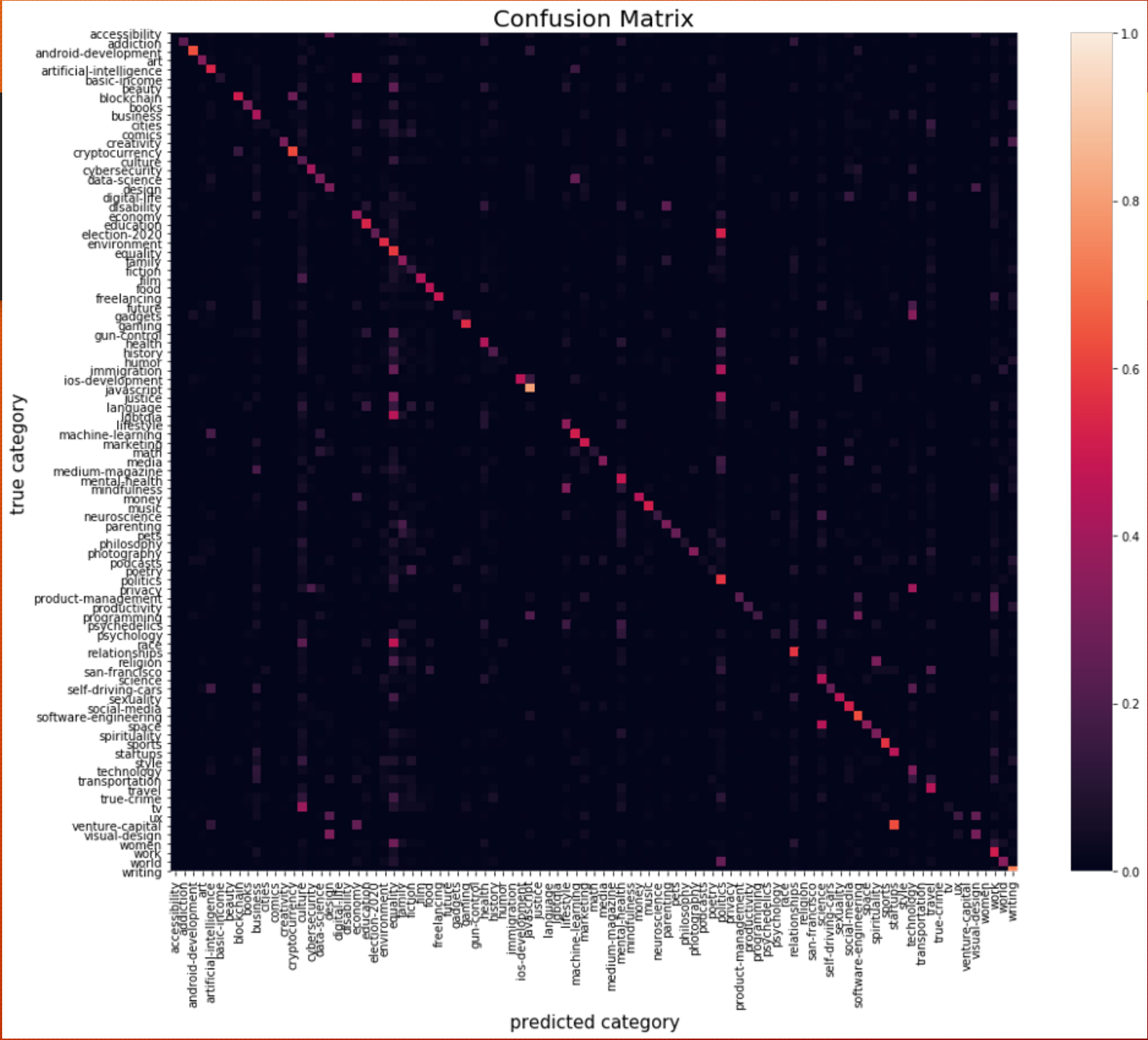
Multilayer perceptron network



To achieve an average precision of 90%, the decision threshold should be more than at least 0.99. The average recall is about 0.5% at that threshold.

Multilayer perceptron network

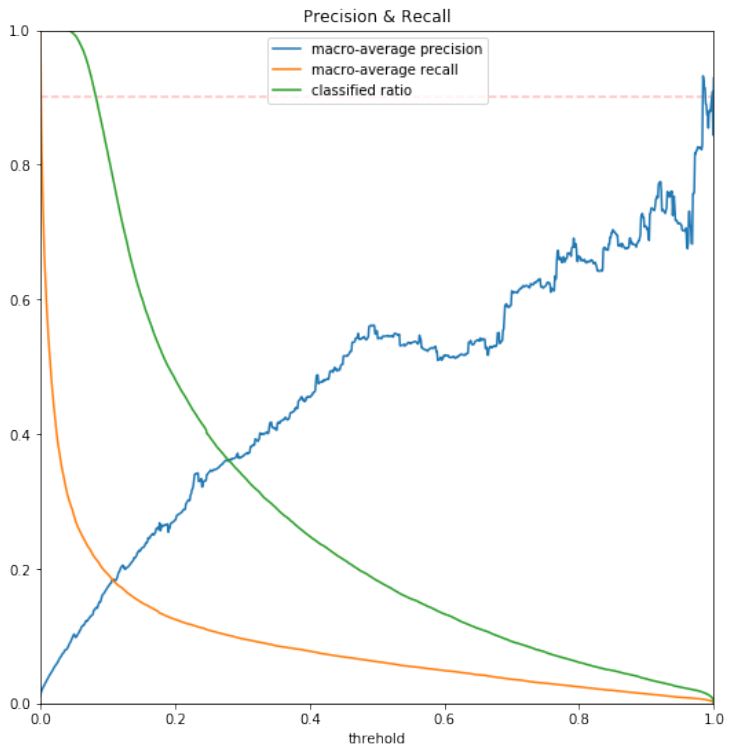
- Confusion matrix
- *privacy* misclassified as *technology*
- *basic-income* misclassified as *economy*
- *election-2020* and *immigration* misclassified as *politics*
- *lgbtqia* and *race* misclassified as *equality*
- *venture-capital* misclassified as *startup*
- *space* misclassified as *science*
- …

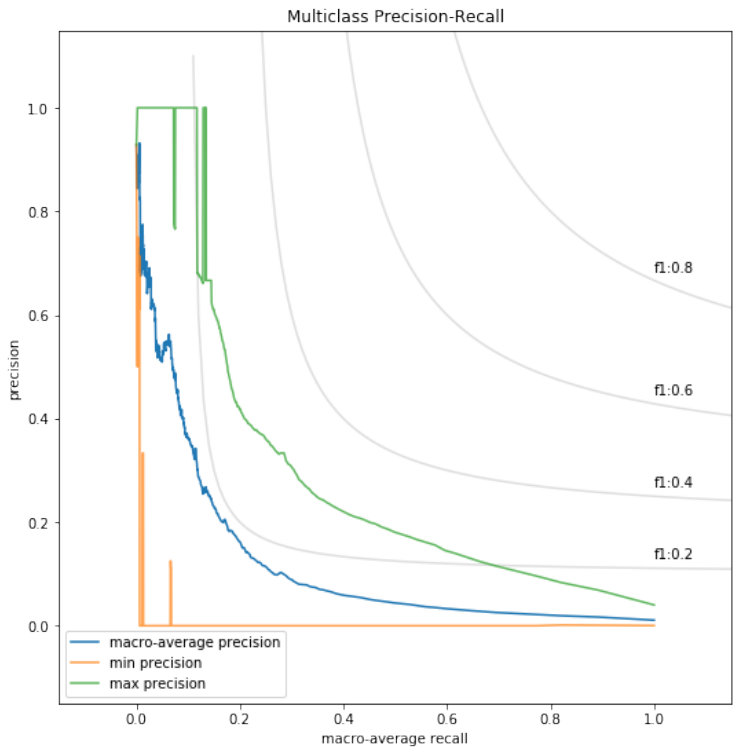


Recurrent network

- 20% test set, 80% training set
- Continuous bag-of-words model with 6-word-wide context, 128d word2vec embeddings
- Architecture:
  - 256-unit recurrent layer
  - 256-unit recurrent layer
  - 256-unit perceptron layer
  - 256-unit perceptron layer
- accuracy on test set: 22%, average multiclass precision: 22%

Recurrent network

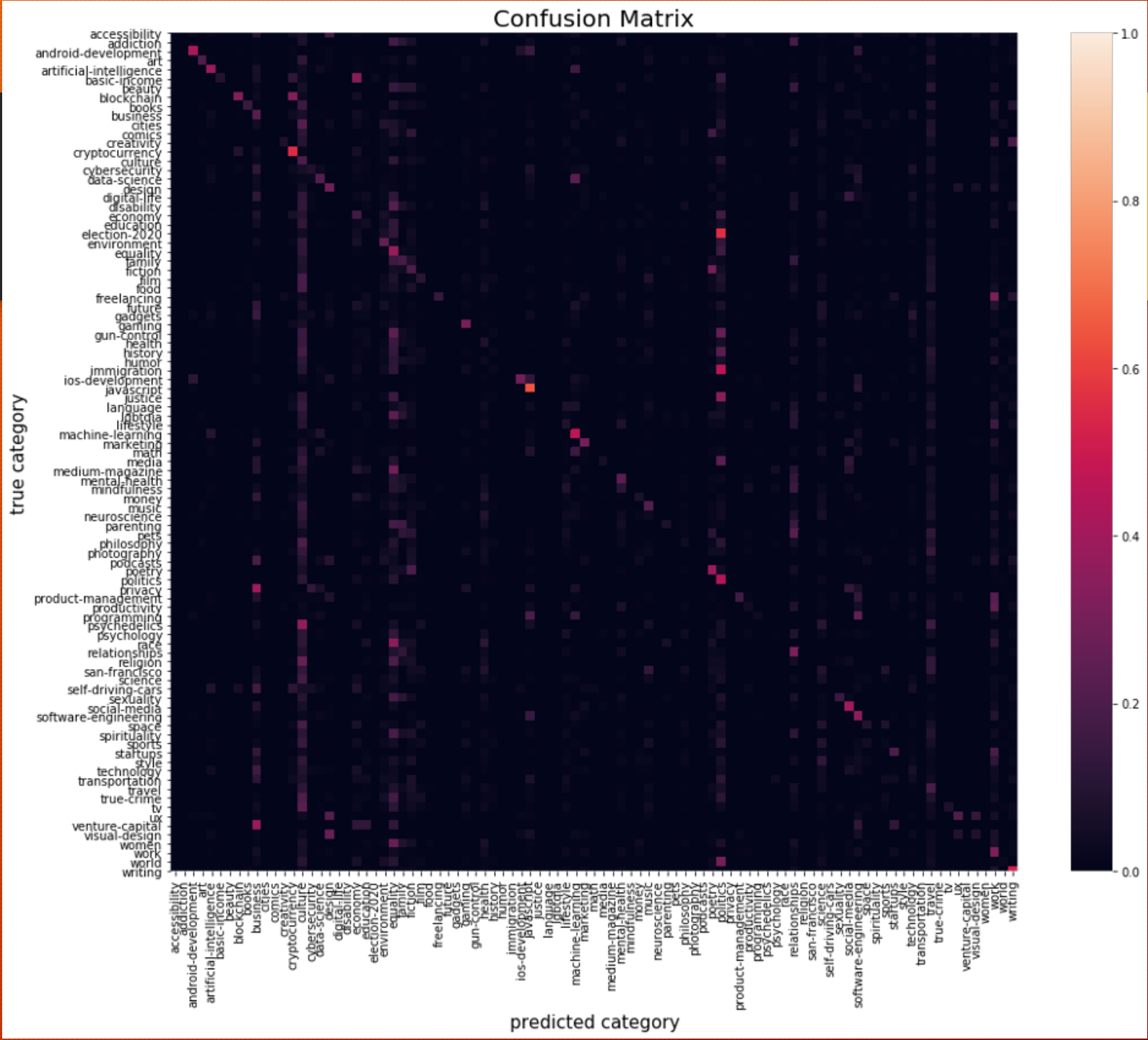




To reach a 90% average multiclass precision, the minimum threshold should be 0.98; the corresponding average multiclass recall is 0.6%.

Recurrent network

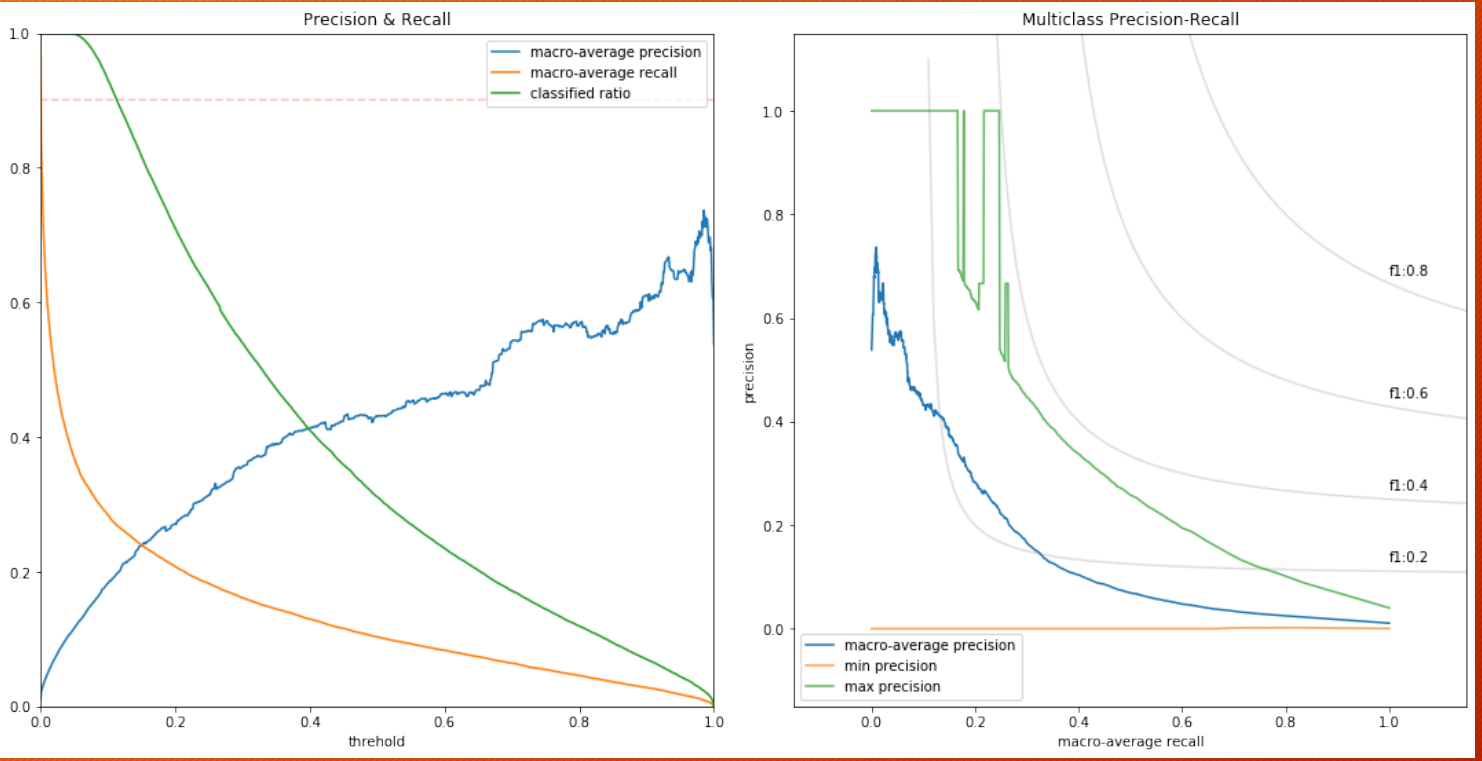
Confusion matrix



Long short-term memory network

- 20% test set, 80% training set
- Continuous bag-of-words model with 6-word-wide context, 512d word2vec embeddings
- Architecture:
  - 512-unit LSTM layer
  - 512-unit perceptron layer
- accuracy on test set: 29%, average multiclass precision: 26%

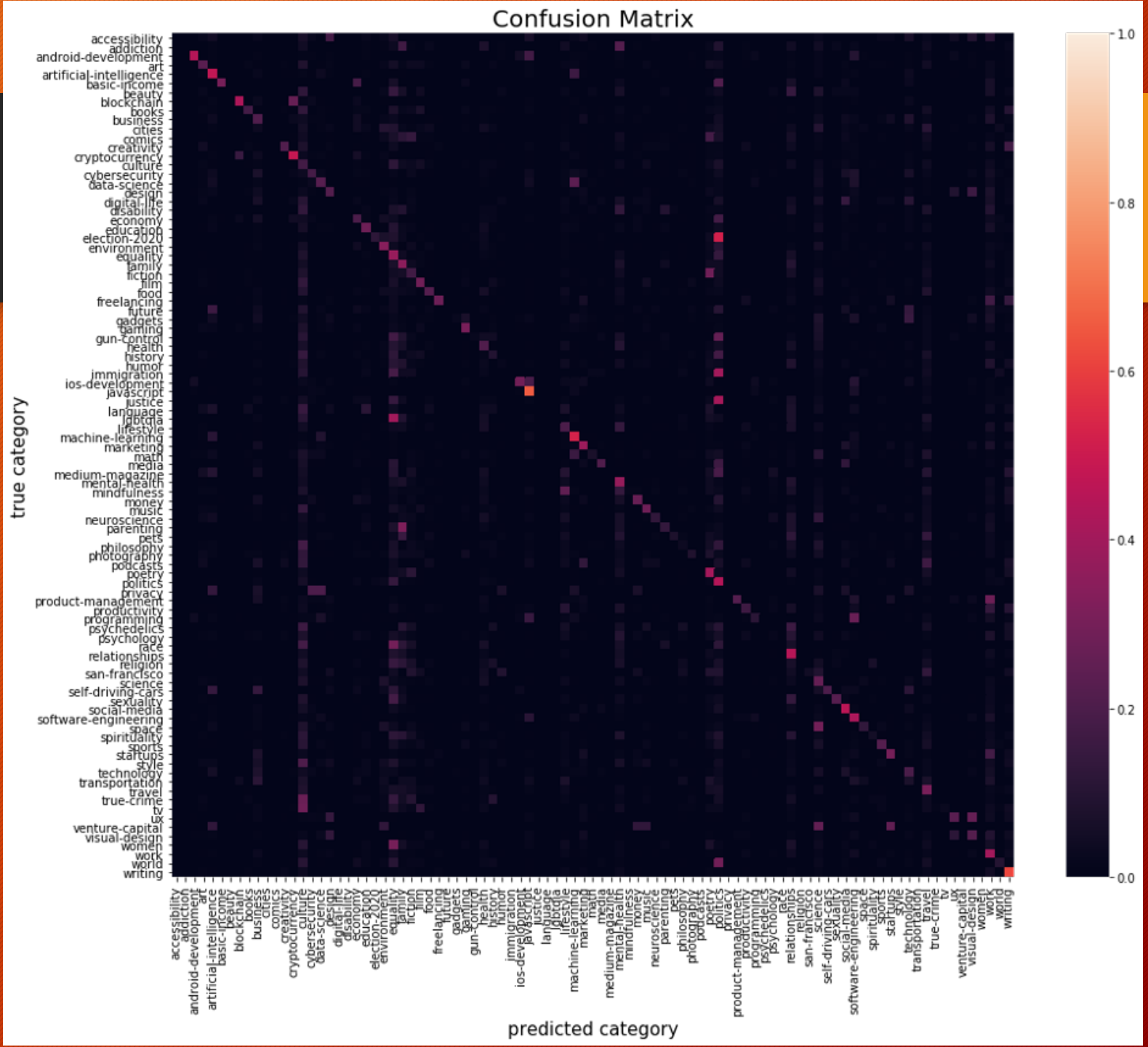
Long short-term memory network



The classifier was unable to reach high precision.

Long short-term memory network

Confusion matrix



Model evaluation

Comparing models

Recommendation system

Optimal model

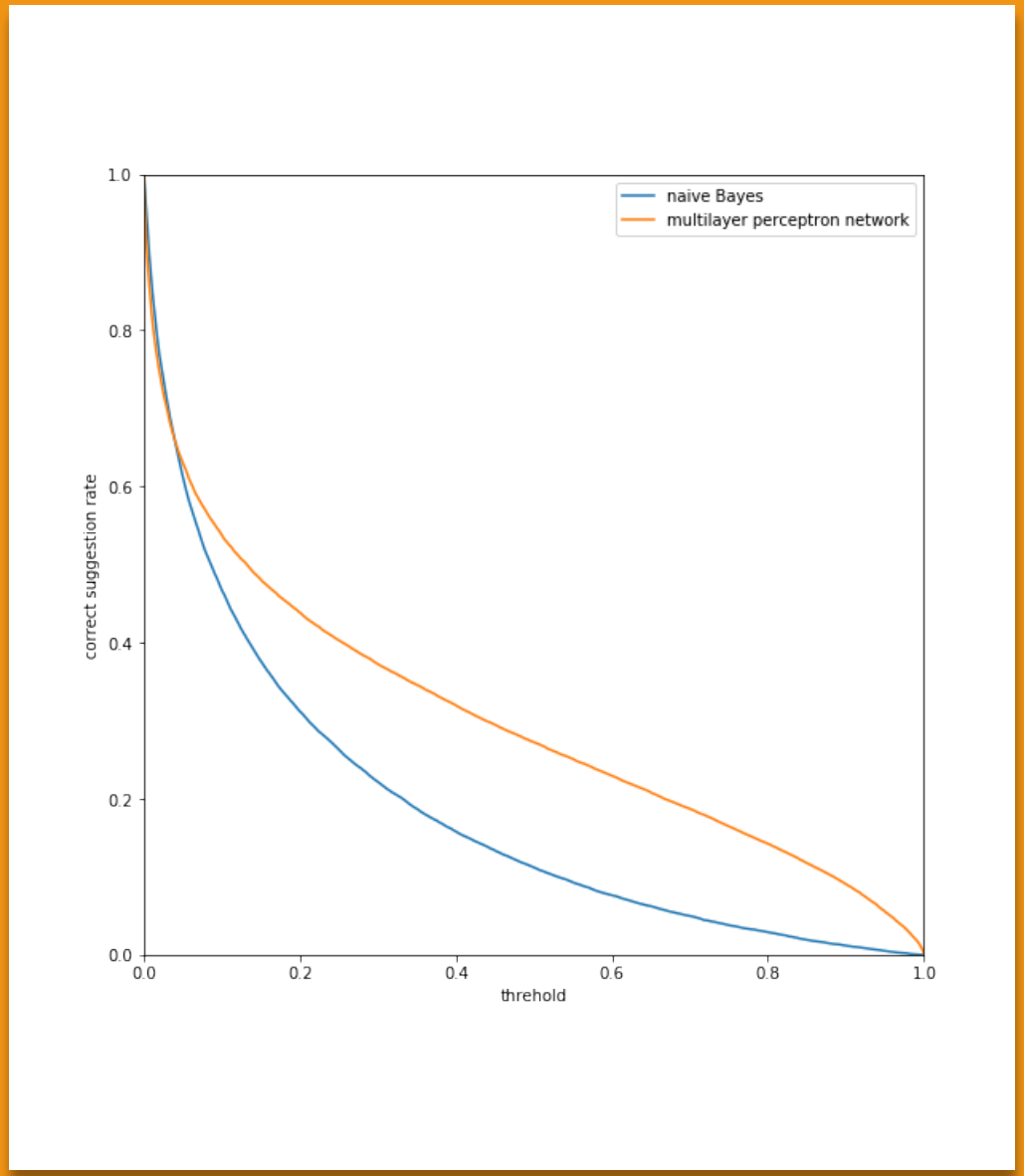
- The naïve Bayes and multilayer perceptron network models deliver better accuracy than the recurrent network and the LSTM network
- Inferior performance of the recurrent network and the LSTM network may be due to titles being too short
- Between the naïve Bayes model and the multilayer perceptron network, the former demonstrates better precision, but the multilayer perceptron network excels in terms of recall and the number of titles that got classified.

Building a recommendation system

- Given a probability threshold for the chosen classifier, all themes with predicted probability higher than the threshold should be suggested as possible choice for the user to choose from.
- A correct suggestion rate can be calculated for the proportion of titles whose suggested themes include the real theme.

Recommendation system: naïve Bayes vs perceptron

- For reasonable working thresholds (>0.1), the correct suggestion rate is significantly higher for the multilayer perceptron network
- At threshold of 0.25
  - The naïve Bayes classifier makes a decision 48% of the time, and the correct suggestion rate is 26%.
  - The multilayer perceptron network makes a decision 92% of the time, and the correct suggestion rate is 38%.



Recommendation system

- **Example:**
- Given the title 'Cheery-picking and False Credibility: A Commentary on the Anti-Vaxxer Media Strategy.', the true category is *social-media*.
- Thresholding at a probability of 0.1, the Multilayer Perceptron Network recommends 2 category/ies: *media*, *social-media*.

Conclusion

- This project demonstrates multilayer perceptron network is a feasible model to build a recommendation system that suggest themes for articles on an online publishing platform based on titles of the articles.
- A similar system can be built for business where many departments are involved. A facilitative algorithm can recommend to agents the possible department(s) a request ticket should be sent to.
- Such recommendation system should boost human agents' operation efficiency.