



Medical Text Classifier

Built Differently by: “The Natural Language Prodigies”

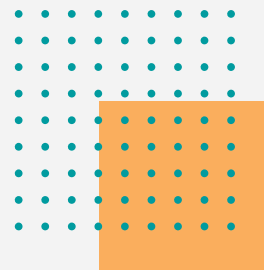




Table of contents

01

Business
Problem

02

Approach

03

Modeling

04

Evaluation

05

Conclusions





Business Problem

- Physicians overworked and overwhelmed
- Lots of research to read
- Our model will point them to the ***most relevant*** research papers by field of medicine:
 - Cancer
 - Digestive
 - Neuro
 - Heart



26.7 hours

Of clinical work per day physicians would need to do to provide guideline-recommended care¹



68%



Of physicians feel overwhelmed by the amount of information they need to review to stay current¹



3.5 years

The rate at which medical knowledge is doubling¹

1. HealthcareFinanceNews.com ([link](#))



Text Cleaning Process



Set up Base Model



Run Machine Learning Models



Model Evaluation

- Logistic Regression
- Gradient Boost
- Stacking Classifier
- Neural Networks
- Decision Tree
- Random Forest
- xgboost



Finalize the best Model



- Dropping Class 5
- Adding LSA

Modelling Approach

Cleaning Process



Define function to remove punctuation, change to lowercase and separate words

01

Tokenize

Filter

02

Use NLTK's stopwords list to filter out superfluous words from our lists of tokens

Lemmatize rather than stem to prioritize accuracy and dataset is not huge

03

Lemmatize

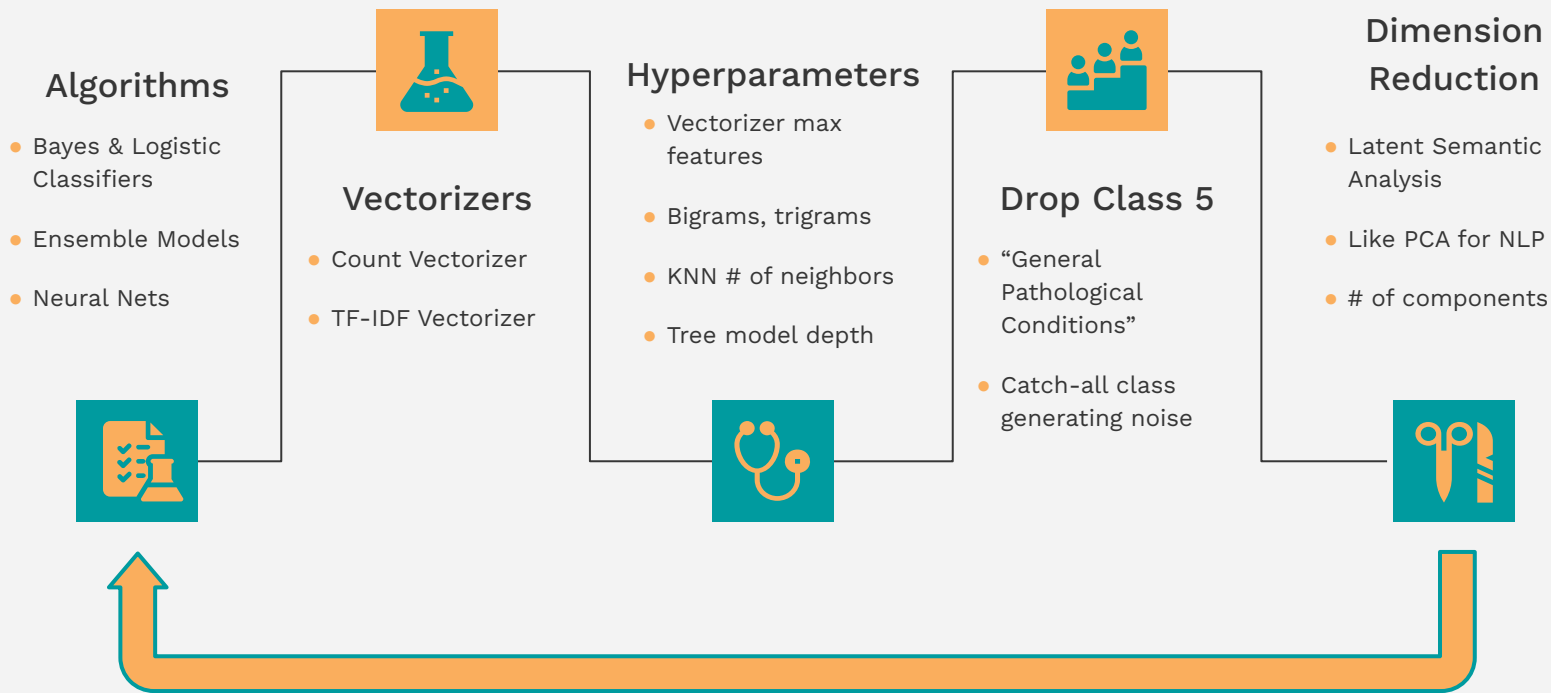
Common Word Removal

04

Identified frequent words common to all classes and removed them



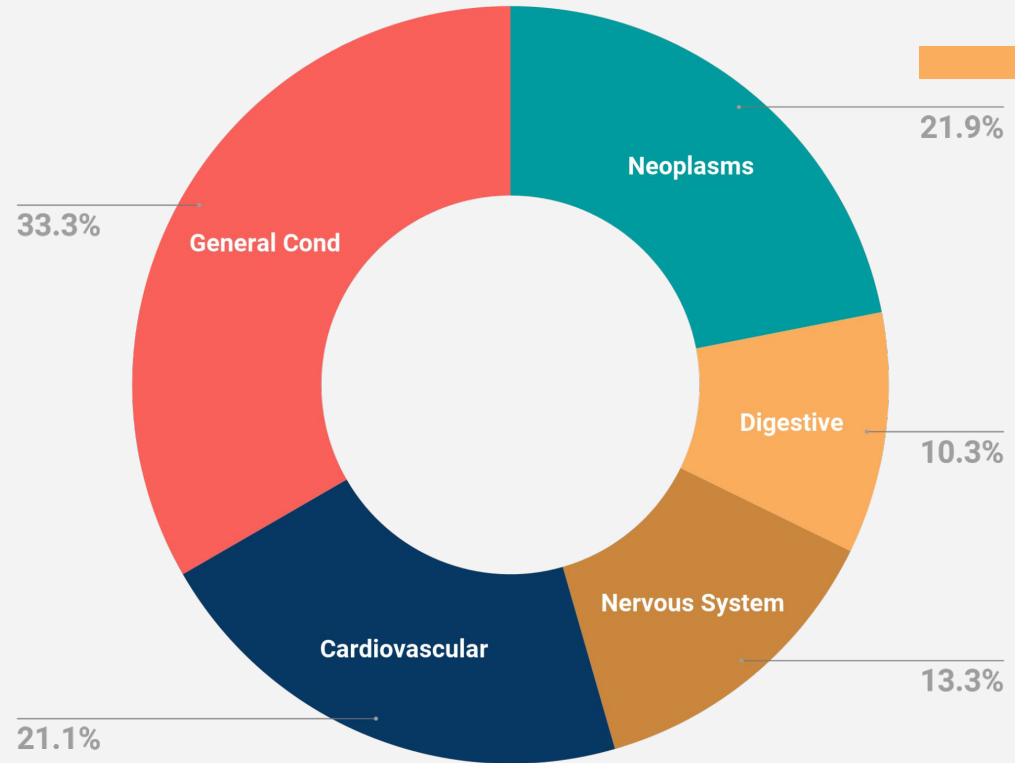
Iterative Process





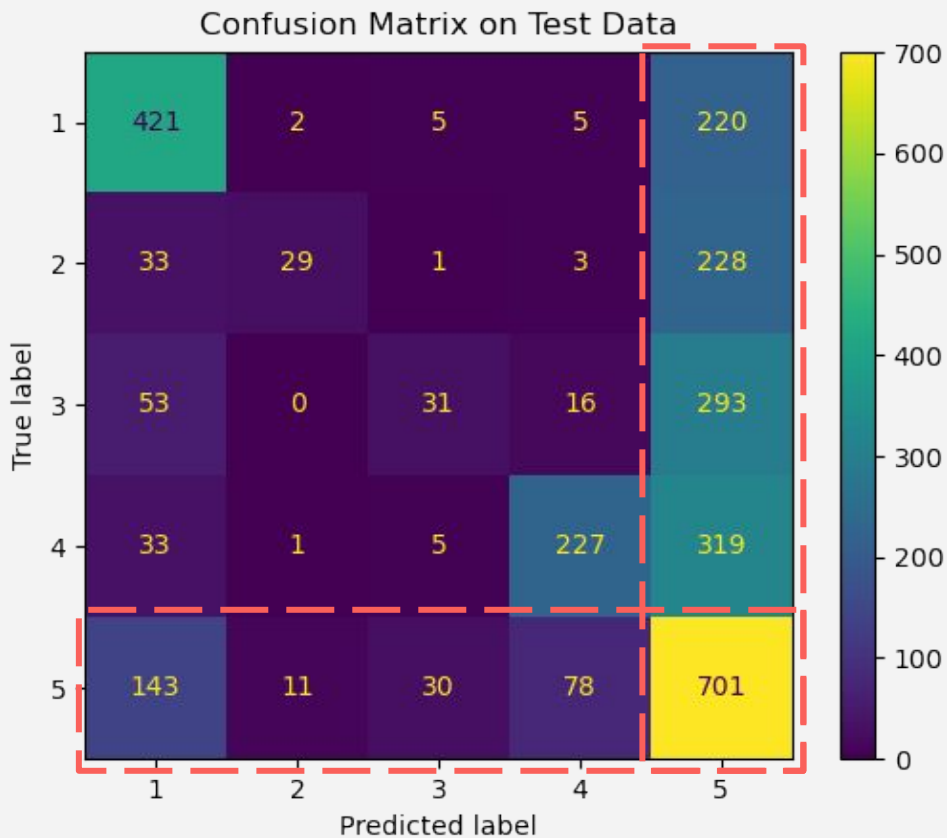
Class Distribution

Neoplasms	3,163
Digestive	1,494
Nervous System	1,925
Cardiovascular	3,051
General Conditions	4,805

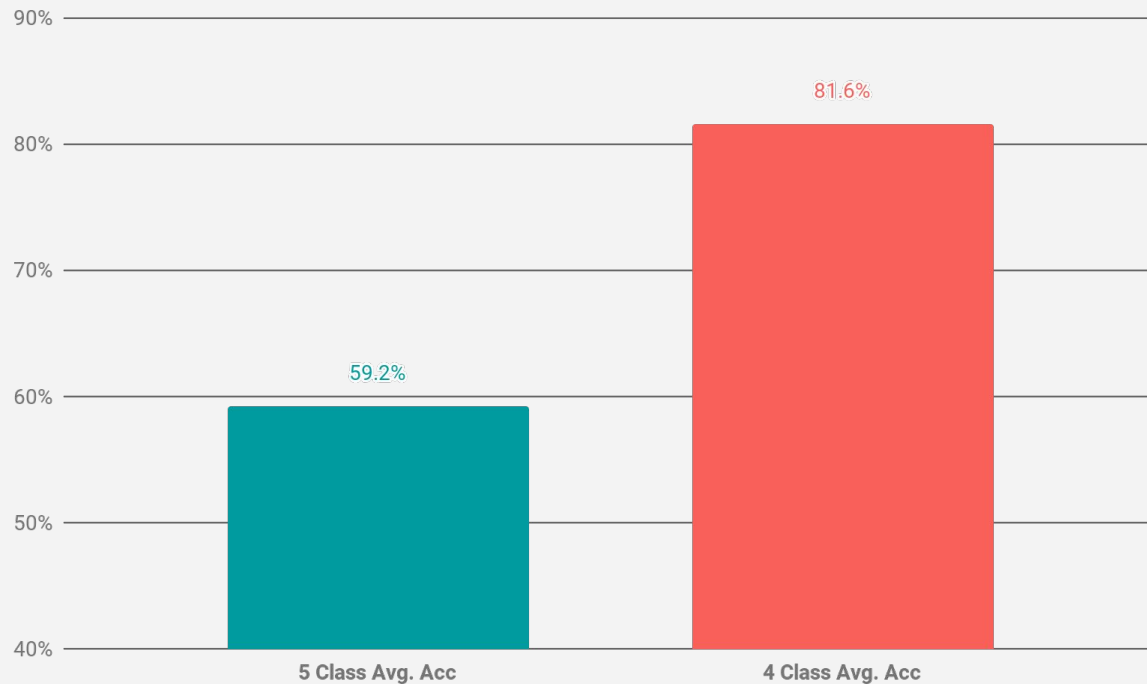


Confusion Matrix for Five Class KNN

1. *Neoplasms (Cancer)*
2. *Digestive diseases*
3. *Nervous system diseases*
4. *Cardiovascular diseases*
5. **General pathological conditions**



5 Class vs. 4 Class



22.4%

Increase in
Accuracy



Hyperparameter Tuning

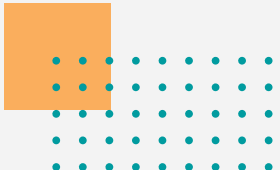


Vectorization

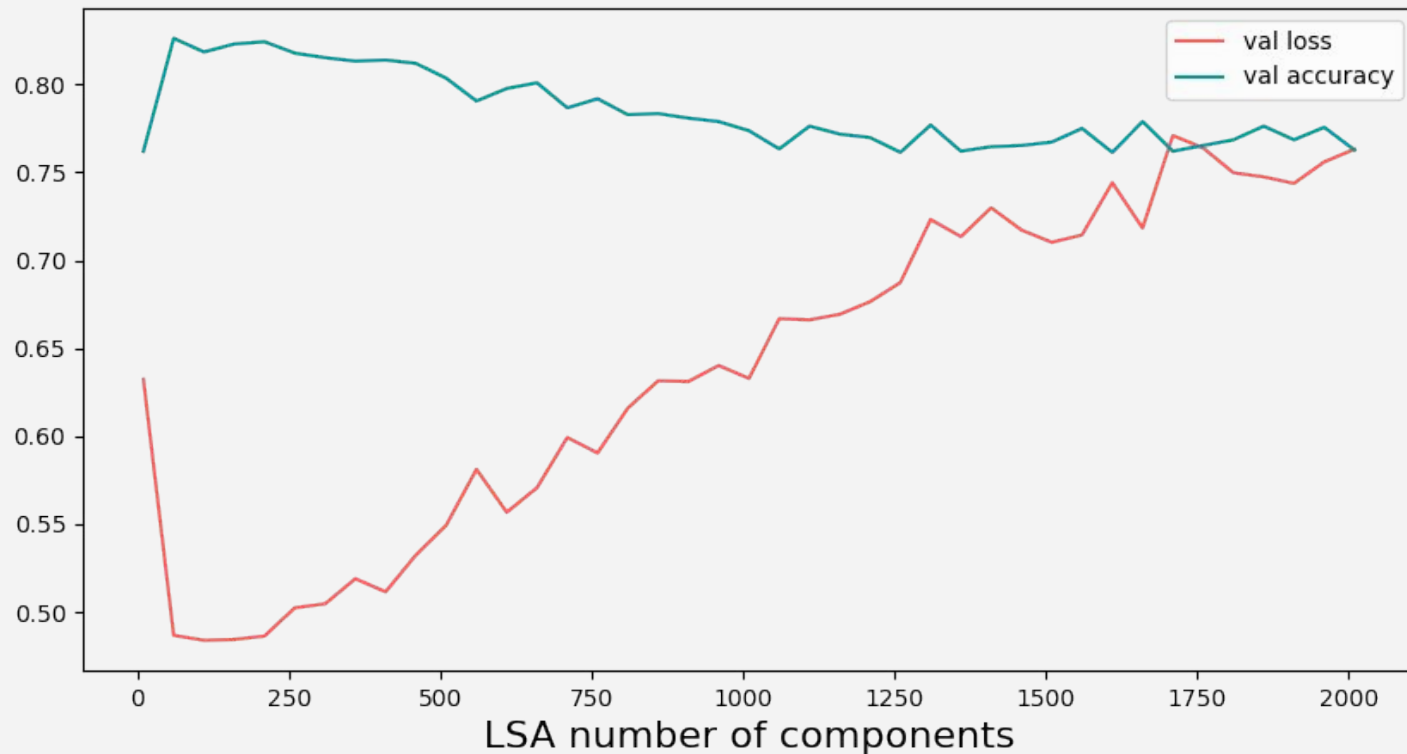
- Count
- TF-IDF

Ngram range

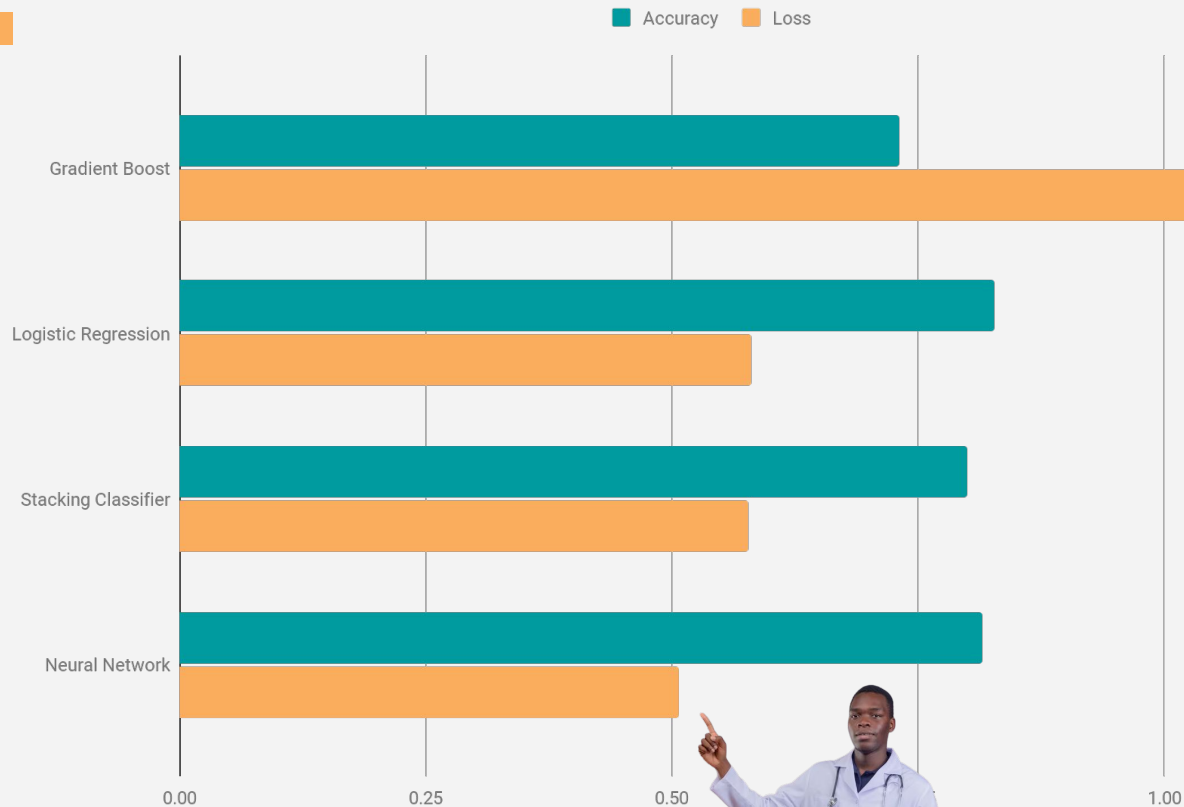
- Unigrams
- Bigrams
- Trigrams



Latent Semantic Analysis¹



Top 4 Models

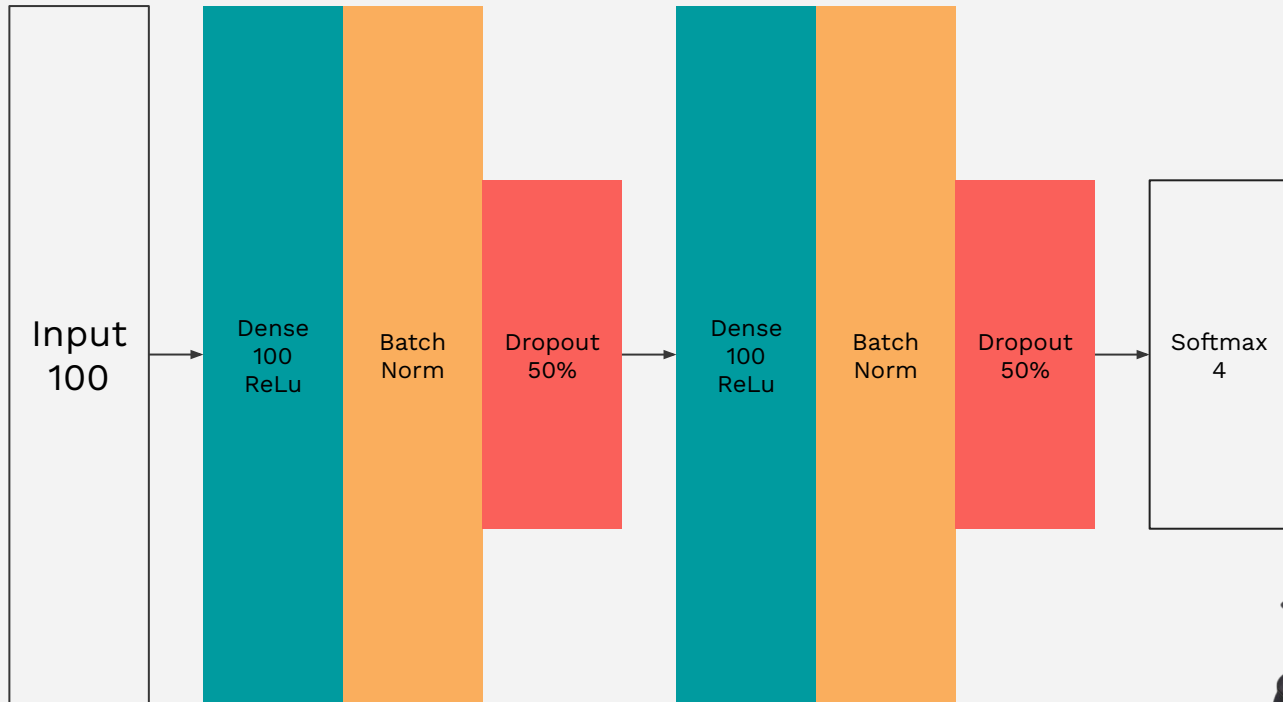


0.507

Neural Network
Log Loss



The Disease Deducer



Params:

- TF-IDF
- LSA = 100
- Adam optimizer
- Batch size = 32
- Early stopping with patience of 5



Conclusion

- Neural Nets are good
- We need more data
- We're ready to sell our model to Google or whatever



Next Steps

- Other possible applications
- Training on more data
- More classes
- Improve accuracy using Word2Vec
- Sequential modeling



Thanks! Questions?



Tristan Trechsel
tristantrechsel@gmail.com
Github: @ttrechsel
LinkedIn: /in/trechsel



Nick Kai
nhknicholas@gmail.com
Github: @nihkai
LinkedIn: /in/Nihkai



Heath Jones
jimmyhj9@gmail.com
Github: @heefjones
LinkedIn: /in/heefjones



Yasitha De Alwis
ydealwis@gmail.com
Github: @yasiSriLanka
LinkedIn: /in/yasitha-de-alwis