Document Vectorization and Predictive Classification of Analytic Philosophy Texts

An examination of Count Vectorization, TF-IDF, with Random Forests and Convolutional Neural Networks

Dave Lobue
Northwestern University
MDS 453: Natural Language Processing
October 20, 2020

Abstract

Document retrieval and classification in natural language processing has reached new levels of sophistication due to advances in word quantification algorithms like TF-DF and word vectorization (e.g., Doc2Vec, word2vec). The following research compares two popular approaches TF-IDF and Doc2Vec performance versus traditional expert based document categorization on machine learning classification tasks using a corpus of texts from the Stanford Encyclopedia of Philosophy. The full corpus includes texts focused on the philosophies of language, mathematics, logic, and mind, which all are approached from the basis of analytic philosophy (predominantly late 19th, early 20th centuries). Given the narrow yet deep focus of these texts, this project seeks to identify the best method of data pre-processing to enable classification algorithms to achieve the highest F1 score across six categories within this subject area. Measurement of F1 scores was conducted using predictions from random forest and Convolutional Neural Network (CNN) approaches.

Introduction

The purpose of this research is to identify the most effective approaches to text classification on a very abstruse, yet focused corpus. The analytic philosophies focusing on language, mind, logic, and mathematics are unique in their objectives but rely heavily on each other in formulating the basis of individual ideas and explanations. As a result, there is notable overlap between these texts despite their being deep subject-specific informational content within sub fields. Finding the optimal approach to text classification for this focused corpus requires identification of key terms and their combination that most closely aligns with narrow concepts. In many cases, these philosophies are dependent on logical or mathematical based proofs to validate theories of mind, language, cognition, or knowledge. Making this task more challenging is distinctions between concepts such as ‘knowledge’, ‘mind’, ‘understanding’, or ‘comprehension’ rely on a broadly overlapping base syntax to deduce philosophical arguments.

This type of research benefits the field of analytic philosophy and academics pertaining to natural language processing due to the narrow focus of topics within the corpus. The classification of philosophical texts using machine learning provides insights into the subtle nuances at the intersection of language, logic, mind (knowledge), and mathematics. It would not be unreasonable for an average reader to read these documents and be without a definitive perspective as to which of the categories listed above a text would most likely fall within. This leads to the second benefit of this research within academics, where machine learning classification can provide insight into subtle nuances across texts as to the probability each document belongs within language, logic, or cognition, for example. Some texts may be labeled as a philosophy of language but contain expansive logical formulas representing words as symbols. This example may result in a classification prediction of 30% logic, 20% mathematics, and 50% language, which can help guide a prospective reader who may be searching for a document leaning more to 40% mental cognition and 60% language focus.

Literature Review

There are no examples of this specific type of research known to the author, although top search results from arXiv.org using queries: “philosophy of language”, “language processing philosophy texts”, “classification of philosophy” within the CS ‘artificial intelligence’ and ‘computation and language’ categories yields papers on “Computer Science and Metaphysics: A Cross-Fertilization” (Kirchner) and “Examining the rhetorical capacities of neural language models” (Zhu). However, these focus more on computational approaches to traditional philosophical problems (e.g., resolving Plato’s proofs and Goedels ontological arguments) using theorems and not specifically ML coding (Kirchner), and language processing models interpretation of rhetoric using Rhetorical Structure Theory (RST).

Methods

To comparatively evaluate the text classification performance, the corpus was prepared in advance removing stopwords (NTLK ‘english’), omitting all punctuation, word frequency (minimum of 3 occurrences) within each document, and lowercase, non numeric filters. The corpus was originally viewed overall from an 'expert analyst' perspective to identify key terms, phrases, and prevalence across documents. A total of 6 distinct categories were identified using 55 keywords based on frequency and consistency across documents within the corpus: (1) logic, (2) mathematics, (3) language, (4) mind, (5) ontology, (6) ethics. The original corpus of 1,029 documents was extracted from the Stanford Philosophy of Encyclopedia using a web crawler retrieving article text from links 2 levels deep originating from the article entry of Ludwig Wittgenstein. Since the links included were based on related content, this ensured the starting corpus remained focused on closely related philosophers and analytic philosophies. Further narrowing of the corpus was based on mentions of the 55 top keywords across the 6 categories, where at least one of these keywords had to be in the top 5 frequency of each document. This resulted in a final corpus size of 546 documents with the following distributions:

Documents by Classification Group:					
Mind: 141	Logic: 84	Ontology: 75	Language: 72	Mathematics: 66	Ethics: 44

The corpus was then randomly split using scikit-learn train test split function into a training set of 446 documents and test set of 100 documents. Here, the three methods of data preparation for analysis were conducted: (1) Analyst Judgment: document text converted to numerical frequency vectors using scikit learn CountVectorizer function filtered for the vocabulary of the most relevant keywords. (2) TF-IDF: scikit-learn TfidfVectorizer fit on the full text set with a limiting vector vocabulary of top 2,893 features. This vector length also ensured all words listed in approach 1 were included for a fair comparison. (3) Neural network embedding using Doc2Vec from gensim package with comparison vector dimensions trained on 50 epochs each.

Lastly, machine learning approach of random forest from scikit-learn and deep learning Convolutional Neural Network classification algorithms were run to provide a comparison across data setup approaches as well as model prediction performance. The same random forest classifier was used across all 3 data pre-processing methods which included 100 estimators with max depth of 10. To consistently compare across pre-processing approaches, a factorial design comparing vector lengths of 50, 150, and 200 was run for the count vectorizer, TF-IDF, and Doc2Vec methods. A single CNN was run to provide an additional point of comparison, with the vocabulary size of 2,893 (to ensure all top key words identified from the analyst approach were included) consisting of a design with layers of: Conv1D(128,8), MaxPooling1D(2), Flatten, Dense(64), Dense(6) layers with softmax activation, categorical cross entropy loss and adam optimizer for 10 epochs.

Results

Each model was evaluated on F1 score incorporating harmonic mean of precision and recall across the 6 classification groups. The top performing model was the analyst identified keywords random forest with a F1 score of 0.778. The modified vocabulary (min 50 occurrences across entire corpus), TF-IDF, and Doc2Vec 50 Dim were all similar in performance, with F1 scores of around 0.65. Lastly was the basic CNN which scored 0.62 F1 score. **Full results are included in the appendix.*

In terms of factorial design comparison of the count, TF-IDF, and Doc2Vec vectors, higher dimensionality typically resulted in lower performance. Limiting vector dimensionality severely hindered the count vectorizer and TF-IDF approaches when compared against the full vectorization of the corpus as a whole. Count vectorizer and TF-IDF only reached a F1 score of .30 and .37, respectively when restricting vectors to 50 dimensions*.

Conclusions

It is possible that bias was introduced with the analyst based approach, since top keywords were used to both filter the corpus and define individual categories. As a result, this should be considered primarily from a benchmarking perspective and primary focus given to the machine and deep learning approaches. With regards to the CNN, the model quickly reached training accuracy of 100% at 10 epochs, suggesting the model was quick to fit and thus overfit. It is likely that a more developed model introducing dropout or modification of layers, individual neurons may yield improved results.

In summary, dimensionality plays an important role regardless of the data pre-processing approach used.   Focusing on top keywords in isolate can return strong scores, but this also requires the greatest level of manual effort on behalf of the researcher. The four approaches performed comparably and it is expected that further refinement of random forest or CNN model parameters can improve prediction accuracy further.

References:

Kirchner, Daniel, Benzmuller, Christoph, Zalta, Edward. Computer Science and Metaphysics: A Cross-Fertilization. August 11, 2019

Zalta, Edward. Stanford Encyclopedia of Philosophy. The Metaphysics Research Lab, Stanford CA. https://plato.stanford.edu/cite.html

Zhu, Zining, Abdalla, Mohamed, Rudzicz, Frank. Examining the rhetorical capacities of neural language models. October 4 2020

Appendix

Top Performing Models:

Corpus Preprocessing*	Classification Model	F1 Score
Analyst Keyword Count Vectors	Random Forest	0.778
Modified Vocabulary Count Vectors	Random Forest	0.659
TF-IDF Modified Vocabulary	Random Forest	0.654
Doc2Vec 50 Dim	Random Forest	0.651
Keras Tokenizer w/Padding	Convolutional Neural Net	0.62

*Full factorial design of 50, 150, 200 Dim for Count, TF-IDF, and Doc2Vec available in appendix

Factorial Design Across 50, 150, 200 Vector Length Dimensions:

Corpus Preprocessing	N Dim	F1 Score
Count Vectorizer	50	0.303
	150	0.1
	200	0.128
TF-IDF	50	0.365
	150	0.112
	200	0.083
Doc2Vec	50	0.651
	150	0.528
	200	0.583

Analyst Judgment (counts) vs. TF-IDF using random display of 8 top words (of 55) against 4 documents

	Word	Logic	Mental	Proposition	Knowledge	Language	Thought	Proof
Mental Imagery: Imagery in Cognitive Science								
Analyst Judgment	5	0	83*	3	5	16	13	1
Tf-Idf	0.01	0.0	0.14	0.005	0.009	0.024	0.018	0.002
The Language of Thought Hypothesis: Mental Language								
Analyst Judgment	4	4	61*	18	0	9	6	0
Tf-Idf	0.03	0.03	0.48	0.0	0.09	0.06	0.04	0.0
Logic and Artificial Intelligence: Reasoning about Action and Change								
Analyst Judgment	4	44*	0	6	1	16	2	0
Tf-Idf	0.01	0.17	0.0	0.02	0.004	0.06	0.007	0.006
Private Language: The Private Language Argument Expounded								
Analyst Judgment	8	0	2	0	1	32*	1	1
Tf-Idf	0.04	0.0	0.013	0.0	0.007	0.18	0.005	0.008

* Most frequently used word in document