Paperscraper: Multi-label Document Classification used for Empirical Literature Review [1]

Andrew Andrade, Andy Toulis, Baha Nurlybayev

December 7, 2015

Abstract[2]

This paper outlines a methodology for multi-label document classification with applications in aiding literature review. Specifically it outlines the data mining approach taken to understand how Artificial Intelligence and Predictive Analytics has been used in petroleum industry. It outlines a modern data collection, storage and analysis pipeline, topic modeling using latent semantic analysis, multi-label document classification, and association rule mining to associate machine learning techniques with applications in oil and gas. The findings were that while a system could be built to cluster and classify papers, there isn't enough data in the abstracts and metadata of literature to establish concrete associations to use cases in oil and gas.

Introduction

A literature review publication is a scholarly paper which contains a summarization of all the current published knowledge about a specific topic. Literature reviews are very important in organizing the current knowledge about a specific topic, and help define future studies. Specifically, by reading a literature review, scholars are able to draw new and original insights from previously published literature. They may be able to find a fresh and original research questions, identify a gap in the literature or make surprising new connections.

The basis of a literature review is from secondary sources as it does not report new or original experimental work. The process of reviewing literature requires different kinds of activities and ways of thinking[3]. It very time consuming to have to manually search, read and review relevant papers. While data mining might not be able to completely remove the need for manual literature review, it can be used to automate routine work in preparation for insights and decisions in technical and scientific writing.[4]

The authors of this paper have identified a need for a literature review of applications of artificial intelligence[5] and predictive analytics (AIPA) techniques in the exploration and production (E&P) side of petroleum industry. Specifically, there are a significant number

[1] Term paper for MSCI 446: Data Warehousing and Mining taught by Professor Lukasz Golab

[2] **Keywords:** data mining, man-machine symbiosis, literature review, text mining, text classification, text clustering, latent semantic analysis, multi-label document classification, exploratory data analysis, association rule mining, empirical study of literature.

[3] Michael J Baker. Writing a literature review. *The Marketing Review*, 1(2): 219–247, 2000

[4] Joseph Carl Robnett Licklider. Man-computer symbiosis. *Human Factors in Electronics, IRE Transactions on*, (1):4–11, 1960

[5] Loosely speaking, AI in E&P is defined as the capability of machine to mimic or exceed human intelligence in everyday engineering and scientific tasks associated with perceiving, reasoning and acting.

of applications of AIPA techniques used in the petroleum industry outlined on OnePetro.org, but there are no comprehensive literature reviews on the topic.[6] The purpose of this project is to create an analytical method using data mining to aid in a literature review.[7]

Significance/Importance:

This approach is interesting because it can fundamentally change the way researchers perform literature reviews. A data-driven methodology of reviewing literature can enhance researchers' abilities to analyze past publications and prepare them for the future.[8]

On the petroleum engineering side, AIPA has been used in the industry for over two decades, Techniques have not yet been consolidated as standard solutions and most applications are case studies or pilot projects. Any engineering practitioner or manager in the upstream side of the oil and gas industry could benefit from understanding how AIPA techniques have been used for decision-making.

Related Work

Multi-document Summarization for Literature Review:

Extensive research has been done in the field of multi-document summarization and is the result of many research publications.[9] Specifically, for multi-document summarization for literature review, studies have been done on a macro-level discourse structure of literature reviews,[10] how information is selected and transformed, a framework for multi-document summarization and on information selection from cited papers in literature review writing. These works are making developments in imitating human literature review writing. All of these studies build the framework for focusing on the auto-generation of multi-document summaries[11] in the form of a literature reviews.[12]

AIPA techniques used in E&P Literature Review:

An attempt has been made to informally review the uses of the techniques through the results of a technical survey which outlines SME's understanding of AIPA techniques (such as machine learning, genetic algorithms, neural networks etc.) and their applications to the upstream side of the industry (reservoir simulation, fault detection, production optimization etc.).[13] This state of the art literature also organizes the current state of knowledge in terms of individuals' understandings of the techniques, not in terms of published work. The

[6] César E Bravo, Luigi Saputelli, Francklin Rivas, Anna G Pérez, Michael Nickolaou, Georg Zangl, Neil De Guzmán, Shahab Dean Mohaghegh, Gustavo Nunez, et al. State of the art of artificial intelligence and predictive analytics in the e&p industry: A technology survey. *SPE Journal*, 19(04):547–563, 2014

[7] As such, this paper is not a literature review, rather it presents a methodology for utilizing data mining to aid in literature review.

[8] Literature review is currently a manual process, our approach maintains that insights are generated by humans, computers are used to aid in the process to advance a more cohesive man-machine symbiosis.

[9] Ladda Suanmali and Naomie Salim. Literature reviews for multi-document summarization

[10] Christopher SG Khoo, Jin-Cheon Na, and Kokil Jaidka. Analysis of the macro-level discourse structure of literature reviews. *Online Information Review*, 35(2):255–271, 2011

[11] Kokil Jaidka, Christopher Khoo, and Jin-Cheon Na. *Imitating human literature review writing: an approach to multi-document summarization*. Springer, 2010

[12] Our proposal is different since we are using data mining as a tool to aid the researcher (man-machine symbiosis method) in reviewing a specific topic, not replacing the need for a researcher to read and review publications and write the review.

[13] César E Bravo, Luigi Saputelli, Francklin Rivas, Anna G Pérez, Michael Nickolaou, Georg Zangl, Neil De Guzmán, Shahab Dean Mohaghegh, Gustavo Nunez, et al. State of the art of artificial intelligence and predictive analytics in the e&p industry: A technology survey. *SPE Journal*, 19 (04):547–563, 2014

goal of this study is to build a framework to aid the comprehensive literature review of uses of AIPA techniques in E&P.

Data

All of the data used in this study was collected by scraping onepetro.org[14] and the general data collection, storage and cleaning process is shown in Figure 1.

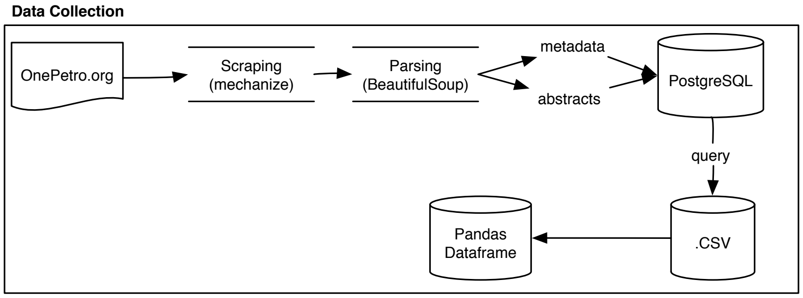


Figure 1: Data collection framework.

Data Collection:

Data was collected through the use of a custom made web scraper script written in `python` using the `mechanize` library.[15] The script attempts to capture all the available meta-data from all the available literature publications on Onepetro.org.[16]

Data Pre-processing and Storage:

From the scraped metadata, `BeautifulSoup` library in python was used to extract the relevant features from the `html` document. The scraped data was stored as `json`, later imported into a PostgreSQL database with the unique `document id` as the key in the key value store.[17] While importing the data into the database, the data was initially sanitized where papers with missing, non-nonsensical or incorrect data were removed.

Data Processing and Analysis Pipeline

After the pre-prossessing, there were a total of 111,420 total papers loaded on the database. The metadata on each paper was then exported to a CSV after a query. The CSV was then loaded into a python dataframe through the use of `Pandas`.[18] In memory analysis provides significantly faster performance, and the use of a virtual

[14] OnePetro.org is an online library (from 18 publishing partners) of technical literature for the oil and gas exploration and production (E&P) industry.

[15] The original script was written by Jonathan "Jay" Estrella while an intern at PetroPredict, but was later modified to better capture hidden metadata from OnePetro.org.

[16] There lacks consistency on the site, so it is significantly difficult to robustly capture all of the meta data let alone every paper.

[17] The choice to use PostgreSQL was due to the fact that the data initially appeared to be too large to perform analytics in memory.

[18] RAM was not a hindrance due to the use of a virtual machine/ server provided by the Computer Science Club at the University of Waterloo.

environment enabled cross-platform collaboration[19] with software dependency limitations. In addition, the data was again cleaned by normalizing all Unicode, dealing with missing values and non-nonsensical text along with removing mislabeled data.[20]

Size of Data:

From the 111,420 cleaned data, 22,250 papers were labeled with Society of Petroleum Engineering (SPE) disciplines.[21] A similar number of papers were labeled with keywords. These two sets had a small overlap of only 100. Keywords were not used in the analysis since many did not have a distinct meaning related to a use case in oil and gas and the frequency of papers with specific keywords was low. For this reason, it was hypothesized that keyword classification would have poor performance, so it was not attempted.[22]

Features of Data:

13 metadata attributes[23] and the abstract of the publication were scraped. Classification and clustering focused on text mining the abstracts, although future analysis could integrate other fields, namely title, discipline and keywords.

Exploratory Data Analysis and Clustering

The first step in better understanding publications in the oil and gas industry was to first do basic visualizations of the scraped data. A plot of the cumulative distribution of literature publications over time is shown in Figure 2. The number of papers published per year is increasing, meaning there is a growing need for performing literature reviews.

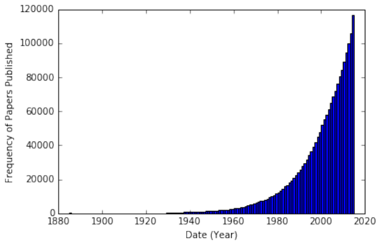


Figure 2: Cumulative distribution of paper publications over time.

In addition, a distribution of abstract length was established to be bimodal, as shown in Figure 3. Due to variation in lengths, using term frequencies alone (without normalizing) as features may not perform optimally.

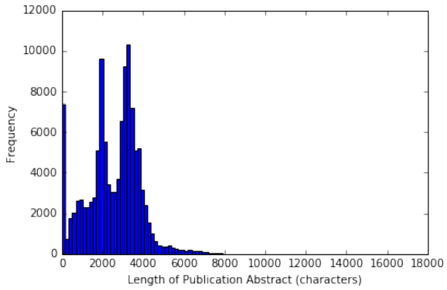


Figure 3: The distribution of abstract length.

[19] The authors of this paper use Windows, Mac and Linux.

[20] The biggest inconsistency we faced was dealing with mislabeled data which had to be labeled manually.

[21] The six disciplines of SPE are Drilling, Safety and Responsibility, Management and Information, Production and Operations, Projects, Facilities and Construction and Reservoir Description and Dynamics.

[22] With the full paper corpus, a good next step could be doing classification using the most frequent keywords.

[23] The title of the publication and its authors, type, DOI, ISBN, copyright, disciplines, keywords, pages, source, date and publisher were scraped where available.

K-means Clustering of AIPA Techniques

The initial approach taken to explore how AIPA was relevant in the petroleum industry was to use of k-means clustering. Each abstract was divided into word n-grams of length 1-3,[24] allowing each paper to be represented as a vector of word frequencies. To explore AIPA, the word counts came from a custom vocabulary the team developed based on our knowledge and research of artificial intelligence, predictive analytics and data mining.

The number of clusters was varied between 2 to 50 and a word cloud of the compiled corpus of the text was constructed and displayed using a wordcloud as shown in Figure 4. The figure shows that an interesting insight that papers about neural networks naturally cluster together, but the results are shallowed out by the noise of unrelated terms. While the papers were able to self-organize through clustering, the overall results were not insightful because of the common words among clusters. To reduce clusters into smaller sets of refined words, we took a latent semantic analysis approach which groups documents by topic.

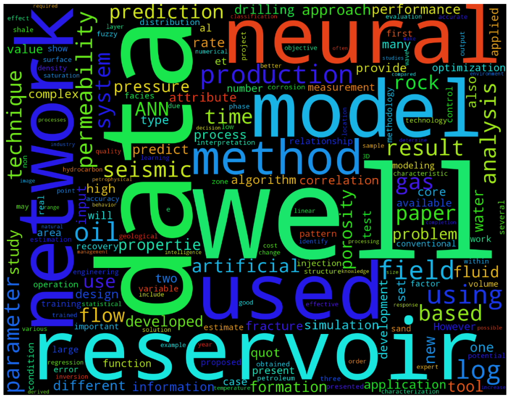


Figure 4: Example of a wordcloud based on a cluster.

Topic Clustering through Latent Semantic Analysis

Latent Semantic Analysis (LSA) is a a technique used "for extracting and representing the contextual-usage meaning of words."[25] The result of LSA can be used to group documents by the meaning of their words. The procedure works by performing Singular Value Decomposition (SVD)[26] on a term frequency-inverse document frequency (tf-idf)[27] matrix to find a reduced sets of words that co-exist frequently.

[24] An n-gram is a contiguous sequence of n items from a given sequence of text or speech, and in our case, we consider up to three grams where each gram is a word.

[25] Foltz P. W. & Laham D. Landauer, T. K. Introduction to latent semantic analysis. pages 259–284, 1998

[26] SVD is a technique used to reduce the dimensionality of a dataset by finding the most important features.

[27] Tf-idf can be used as a statistic to reflect how important a word is to a document. It is proportional to the frequency of a word in a document and is inversely proportional to the frequency in which the word appears in other documents.

SciKitlearn recommends to use a reduced dimension of 100 for text data[28] and larger sets ranging up to 300 words yielded similar results. Even a lower dimensionality of 50 words performed similarly, suggesting a majority of semantic information is captured by a small set of words in the corpus. This was a key insight that informed our classification approach since it meant that important information could be contained in a set of rare words.[29]

Due to the sparseness of a tf-idf matrix on text like abstracts[30], the method used for SVD was SciKitlearn's `TruncatedSVD`, which operates on vectors instead of a full matrix.[31] Re-normalization is neccessary after using this procedure. SciKitlearn's `Normalizer` was used to re-normalize the vectors. Finally, k-means clustering was performed on the resulting 100-dimensional vector space and the number of clusters was varied from 2 to 12. LSA topic clustering far outperformed the previous approach. The top features for 4 clusters are found below, with duplicates across clusters, like "oil" removed:

1. **Cluster 1**: Seismic, rock, wave, method, stress, velocity, analysis, waves, numerical, surface, strength
2. **Cluster 2**: Preview, available, ATCE[32], commercially, OTC[33], technologies, readily, limited, engineers, houston, currently
3. **Cluster 3**: Corrosion, safety, industry, SPE[34], offshore, management, systems, project, process, operations
4. **Cluster 4**: Reservoir, wells, pressure, flow, field, permeability, fluid, fracture, injection, reservoirs, formation

Cluster 3 shows an interesting segment that focuses on safety, projects, management, processes and operations. Cluster 2 is oriented around technology releases and conferences. Clusters 1 and 4 are specific fields and are distinct from each other. Note that these words were not stemmed for human readability, but the final clustering was performed on stemmed words for performance optimization. Furthermore, a custom vocabulary was built to remove duplicates found across sectors and common words such as reservoir, model, pressure and drill that were not adding meaning.

In addition to removing certain words from the corpus to dig deeper for meaning, further improvements were made to the tf-idf vectorizer. Following the insights from classification, to be discussed in the next section, the maximum document frequency was set to 40% such that common words across industry publications were removed. N-grams ranging from length 1 to 3 were included in the to yield further semantic insights based on words that appear beside one another. This range was selected since it performed optimally

[28] SciKitLearn. Truncatedsvd. b. URL `http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html`

[29] When attempting classification using LSA, it was remarkably found that a dimension of 15 yielded the best performance, meaning a very small set of words contain a large amount of information.

[30] Even when filtering words that appear only once and frequent words that appear in 40% of documents, there are still about 179,000 unique words in the corpus of the 111,420 abstracts. A majority of these words appear only in a few documents.

[31] SciKitLearn. Truncatedsvd. b. URL `http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html`

[32] ACTE stands for the Annual Technical Conference and Exhibition.

[33] OTC stands for the Offshore Technology Conference.

[34] SPE stands for the Society of Petrolium Engineers.

during model classification. Interestingly, n-grams did not make it into the top features after performing SVD and clustering, suggesting that SVD trims rare, but potentially important word pairs.

Figure 5 illustrates the top features in a cluster obtained from the final LSA iteration with 10 clusters. The size of words are proportional to their frequency in the document set belonging to the cluster.

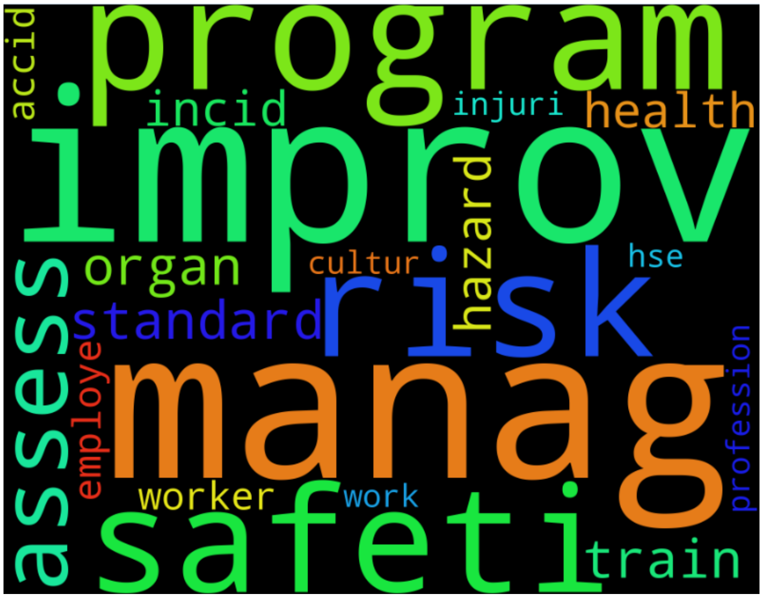


Figure 5: Example of a wordcloud of top features in a document cluster.

Document Classification

Conducting document classification on literature papers creates a multi-label problem. This involves predicting properties which are not mutually exclusive.[35]. A common approach to multi-label classification is through the use of multiple classifiers which learn a discriminative model for each class.

Initial Approach OneVSRest using Linear SVM

As shown in Figure 6, the initial approach taken for multi-label document classification was using the `OneVsRest` function in SciKitlearn which uses a Support Vector Machine (SVM) with a Linear Kernel and a 70:30 split on the training and test data.[36]

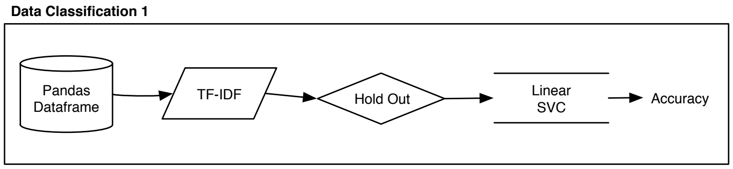


Figure 6: Initial approach to multi-label document classification.

In addition, we used accuracy as the metric to evaluate the performance of the classifier. The classifier showed very high accuracy (around 80%) across all the classes, but upon further inspection[37] it was realized that the classifier was severely underfitting. We later used the Receiver Operating Characteristic (ROC) as a method of visualizing performance, as shown in Figure 7, for the classifier under investigation.

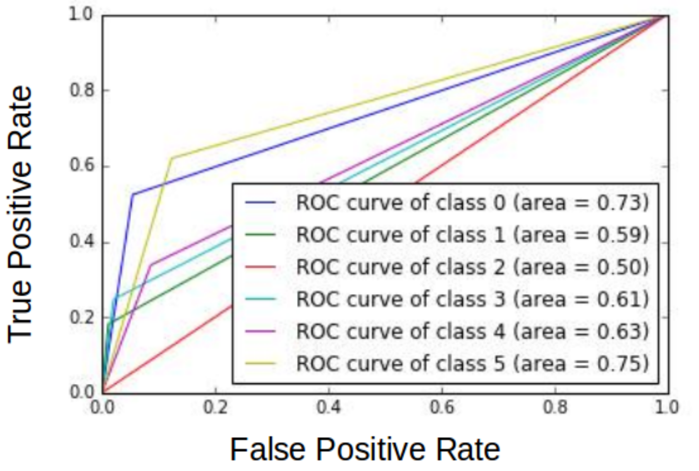


Figure 7: Initial Results of Classification Using a Linear SVM.

[35] For example, in the oil and gas industry, a publication could be under one or more of the 6 major SPE disciplines or none at all.

[36] Initially, the classifier trained on 70% and the performance was tested using the remaining 30%.

[37] We found the Confusion Matrix had no true or false positives. This means the model simply labeled all data as negative.

Classification Optimization

After building and learning from the preliminary classifier, we then made the following optimizations:

1. Hyperparameter tuning using Pipelines and Grid Search
2. K-fold Cross-Validation (CV)
3. Matthews Correlation Coefficient as an evaluation metric
4. SVM using Stochastic Gradient Descent
5. Dimensionality Reduction
6. Random Forests Classifier
7. Hold-out method + K-fold CV for final model evaluation

The overall framework for analysis is shown in Figure 8.

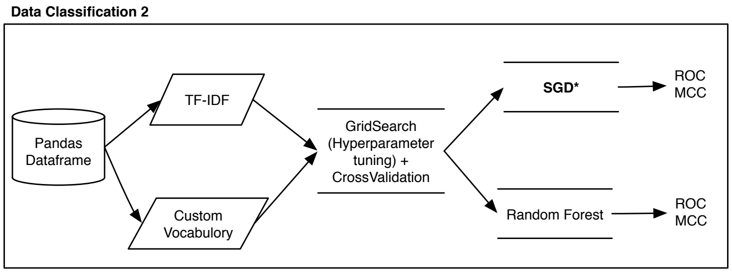
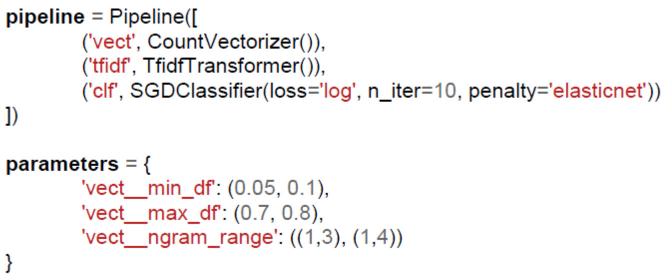


Figure 8: Iterative approach for multi-label document classification.

Grid Search and Hyperparameter Tuning

To automate the analysis, hyperparameter tuning[38] was done using SciKitlearn Pipelines, Parameters and GridSearch. An example of a pipeline used in the gridsearch is shown in Figure 9. This example has three steps: term counting, a tf-idf transform and classifier estimation. Different counting hyperparameters are tested.



```
pipeline = Pipeline([
        ('vect', CountVectorizer()),
        ('tfidf', TfidfTransformer()),
        ('clf', SGDClassifier(loss='log', n_iter=10, penalty='elasticnet'))
])

parameters = {
        'vect__min_df': (0.05, 0.1),
        'vect__max_df': (0.7, 0.8),
        'vect__ngram_range': ((1,3), (1,4))
}
```

Figure 9: Example of a pipeline and hyperparameters used for classifier evaluation.

[38] While a specific classifier determines its own parameters from labeled data, as a data scientist, you provide (hyper)parameters which the vectorizer and classifier uses. For example, the specific features, loss functions or regularization coefficients are all part of model selection done by a human.

Once the parameters and pipeline are setup, SciKitLearn's `GridSearchCV` method can be used to evaluate a combination of all the hyperparameters. The GridSearch documentation[39] offers best practices we followed for hyperparamter tuning using GridSearch.

K-fold Cross-Validation

During the experimentation phase of model pipelining, 3-fold cross-validation, the default for the grid, was used to reduce computation time. Standard 10-fold cross-validation was used in all final models.

In some cases, classifiers were performing significantly poorer on randomly generated folds of the set it was trained on. As an investigation, the training data was split into new random folds after classifier estimation. Plotting an ROC curve for each fold, like the one seen in Figure 10 on a test SGD classifier, showed high variance in performance. After digging into SciKitlearn documentation StratifiedKFold, it was discovered that, by default, the cross-validation method on the grid does not shuffle data. Since our data may have been ordered meaningfully, such as by keyword or date, it needed to be shuffled before being passed into the grid.

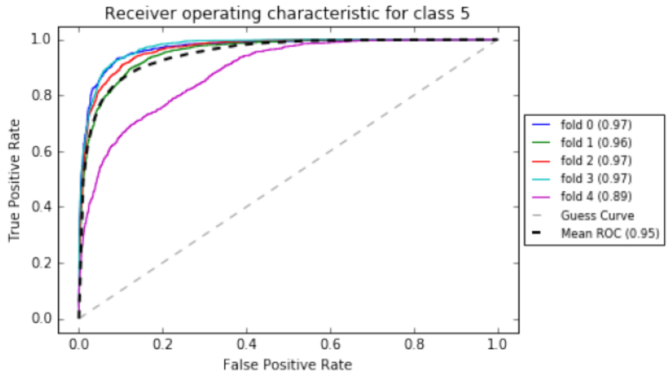


Figure 10: Example of poor performance across a random fold for a sample class and test classifier.

Another issue investigated was that there are very few positives compared to negatives for all classes. To try to balance out the number of positives, an inverse class weighting was used during estimations. The results were not significantly different. Upon doing further research, it was found that the grid automatically performs stratified k-fold cross-validation, meaning each fold is designed to contain a balanced amount of positives and negatives.

Model Scoring Metric

There are many more negatives than positives in each class. As such, true negatives dominate accuracy-based scores. This means that

[39] SciKitLearn. Gridsearch documentation. a. URL `http://scikit-learn.org/stable/modules/grid_search.html`

precision and recall are better metrics to look at for the dataset. The F1 score is an effective metric which combines precision and recall. A refined version of F1 called FBeta is used in practice since it allows for higher weighting of either precision or recall. Another effective scoring technique for unbalanced classes is Matthew's Correlation Coefficient (MCC), which takes into accounts all four quadrants of a Confusion Matrix. MCC was taken as a final scoring metric as it is generally regarded as a balanced measure which can be used even if the classes are of very different sizes and it does not need further tuning like an FBeta score.

SVM using Stochastic Gradient Descent

To speed up the analysis, Stochastic Gradient Descent (SGD) was used instead of a normal Support Vector Machine. This is because SGD is a faster, iterative method for estimating the gradient. It is used in many machine learning and optimization problems. The linear kernel SVM model was replaced by a SGD estimator, yielding significant performance improvements. Classifier estimation was reduced from 20 minutes to 4-5 minutes on the full text.

Dimensionality Reduction

Operating on the tf-idf matrix was computationally expensive due to its sparseness and large dimension. In order to better inform the vectorization process, a unique vocabulary of words contained in the SPE disciplines and sub-disciplines was constructed. This yielded 494 discipline-specific words which could be vectorized. Models were able to perform nearly as well on this vector space, as shown in Figure 11, suggesting that filtering important words from metadata is a useful technique for mining specific topics efficiently.

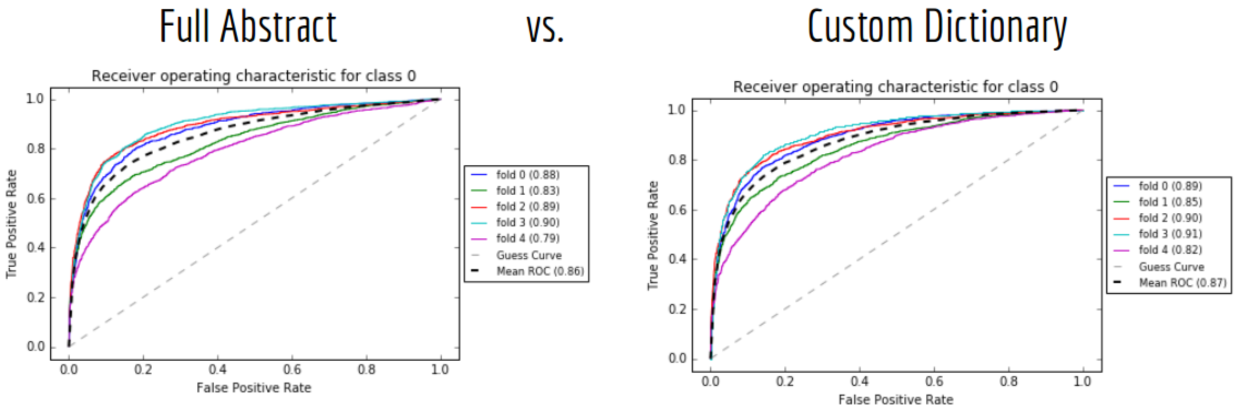


Figure 11: Performance of Full Abstract vs. Custom Dictionary for a sample class.

Random Forests

In addition, the Random Forests classifier was also evaluated and it historically offers very good "out of the box performance".[40] We found that random forests offered very good out of the box performance and built models very quickly (in a couple of seconds) with the reduced feature set. Upon an expansive grid search, we found that we were able to obtain results where the MCC and the area under the ROC curve both approached 1 as shown in Figure 12. As suspected, when using the 70:30 hold-out method, we found that the model was significantly over-fitting since the testing results were a lot poorer than during the training as shown in Figure 13.

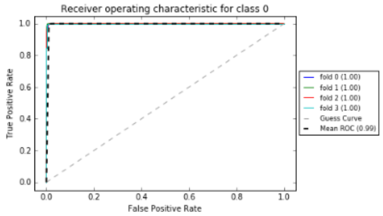


Figure 12: Example of Random Forest Overfitting.

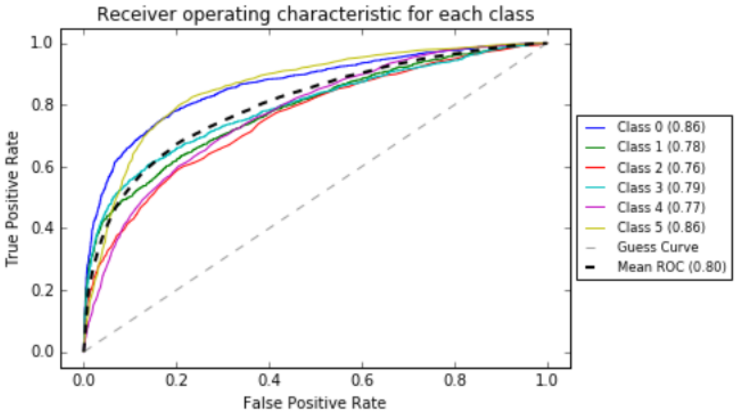


Figure 13: Testing held out data for random forests.

Upon further investigation of the most importance features in the Random Forest classifier, we found that there were common words used as key features across classes. Some of these words matched the results of the initial iteration of LSA, indicating that example words such as reservoir, format, model, fit, pressure, drill and use were examples of words common across multiple classes. These words closely matched the results of the initial iterations of LSA, indicating that they should be removed from the feature set.

[40] As outlined in many blogs, in the early days of Kaggle data science competitions, classification using random forests won without much hyperparameter turning.

Developing a Final Model

Detailed hyperparameter tuning was performed on the SGD classifier since it was not over-fitting like the random forests. Several iterations yielded the following findings:

1. For the count vectorizer, the minimum document frequency that performed best was very small and corresponded to a document frequency of approximately 2 (which is the default). Larger values can be used to improve efficiency.

2. The optimal n-gram range was 1 to 3. It turned out that n-grams have diminishing returns and that 4-grams did not add extra information to the model.

3. Elastic net penalty performed optimally across all classes compared to other available options for SGD on SciKitlearn. The elastic net often outperforms an L1 based penalty when using real world data and is particularly good for high-dimensionality data like a tf-idf matrix.[41] With respect to text mining, elastic net is also useful since it has a grouping effect that causes correlated tf-idf components to either be a part of a model or not.

4. For probability predicting models, the modified Huber loss significantly outperformed Log loss across all models. A simple Huber loss is quadratic for small values of error but linear for large values, making it tolerant to outliers.[42] Log loss takes into account the certainty of classification. It performed well under an accuracy metric, which would cause models to label a majority of data negative with high certainty. Since an MCC scoring metric punishes low recall very strongly, weak performance using log loss may indicate that there is low certainty in positive classification results. Hinged loss performed slightly better in some cases but does not provide probability estimates and therefore was not used for classification.

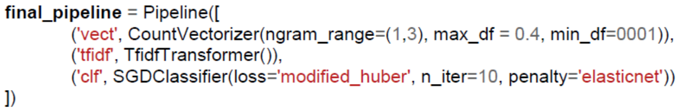
5. A max document frequency of 40% was optimal for four of six classes. 35% and 45% performed slightly better for two classes, but to be consistent and generalized, 40% was used for all classes in the final pipeline.

6. Performing SVD on the tf-idf vector space was not effective, yielding much poorer performance for components of multiple sizes (10, 15, 100, 200, 400).

[41] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005

[42] Other loss metrics, like least squares, are highly sensitive to outliers.

The final pipeline resulting from these insights is shown in Figure 14. It used a count vectorizer to filter n-grams, a tf-idf transformer to re-weigh vectors and a SGD classifier to label data. The number of iterations for SGD was set to 10 as this performed slightly better than the default of 5.



```
final_pipeline = Pipeline([
          ('vect', CountVectorizer(ngram_range=(1,3), max_df = 0.4, min_df=0001)),
          ('tfidf', TfidfTransformer()),
          ('clf', SGDClassifier(loss='modified_huber', n_iter=10, penalty='elasticnet'))
])
```

Figure 14: The final pipeline used to model the data.

The MCC performance of the final model on a randomly selected training set of 70% of the data was the strongest out of all previous SGD models. More importantly, out of all models developed, including Random Forest, the final model performed optimally on the hold-out sample, as seen in Table 1. In fact, for three classes, the model performed better on the hold-out, which is quite remarkable. The average area under the ROC curve was across curves was 0.82.

Table 1: MCC Scores of the final model.

Class	Test MCC	MCC Hold-out
Drilling & Completions	0.60	0.62
Health, Safety, Security, Environment & Social Responsibility	0.46	0.46
Management & Information	0.33	0.36
Project Facilities & Construction	0.50	0.51
Production & Operations	0.41	0.40
Reservoir Description & Dynamics	0.58	0.56

Using the final model, 89170 predictions were made on the remaining unlabelled data, as seen in Table 2. These results were used to explore the relationship between the discipline predictions and AIPA techniques in papers through association rule mining.

Table 2: Predictions by final model.

Class	Number of Predictions	Percent of Total Data
Drilling & Completions	13140	15%
Health, Safety, Security, Environment & Social Responsibility	4230	5%
Management & Information	3050	3%
Project Facilities & Construction	15200	17%
Production & Operations	15300	17%
Reservoir Description & Dynamics	18560	21%

Final Model Evaluation: High False Positives

While the final model offered good performance, the false positive rate was still high. Table 3 shows the Confusion Matrix of the final model for a sample class. The overall difficulty in approximating positives with high confidence is reasonable since manual labeling of the disciplines is a difficult task which is likely incomplete. Additionally, it is reasonable to hypothesize that already labeled papers are less likely to be labeled again into other disciplines despite their partial relevancy. If this is the case, in terms of a multi-label problem, this makes labeled inter-disciplinary data counter-productive for predicting other classes.

	Predicted P	**Predicted N**
Actual P	1333	286
Actual N	806	4294

Table 3: Sample final model Confusion Matrix demonstrating a high false positive rate.

Association Rule Mining

Finally, association rule mining was done attempting to relate the n-gram AI techniques to the SPE disciplines as outlined in Figure 15.

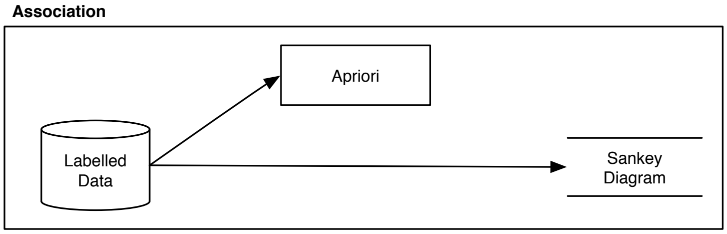
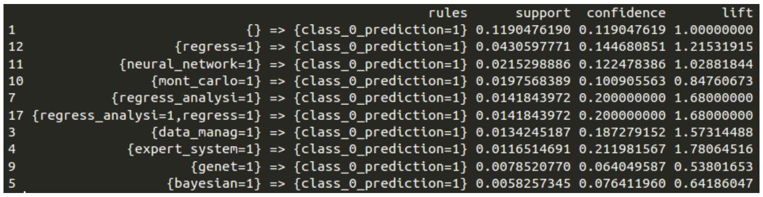


Figure 15: Association Rule Mining Applied to Dataset.

Even when limiting the n-gram AI techniques to a frequency greater than 200, the results of the apriori algorithm yielded very low support and confidence as shown in Figure 16.



```
                                        rules      support  confidence       lift
1                           {} => {class_0_prediction=1} 0.1190476190 0.119047619 1.00000000
12                 {regress=1} => {class_0_prediction=1} 0.0430597771 0.144680851 1.21531915
11          {neural_network=1} => {class_0_prediction=1} 0.0215298886 0.122478386 1.02881844
10              {mont_carlo=1} => {class_0_prediction=1} 0.0197568389 0.100905563 0.84760673
7          {regress_analysi=1} => {class_0_prediction=1} 0.0141843972 0.200000000 1.68000000
17 {regress_analysi=1,regress=1} => {class_0_prediction=1} 0.0141843972 0.200000000 1.68000000
3               {data_manag=1} => {class_0_prediction=1} 0.0134245187 0.187279152 1.57314488
4            {expert_system=1} => {class_0_prediction=1} 0.0116514691 0.211981567 1.78064516
9                    {genet=1} => {class_0_prediction=1} 0.0078520770 0.064049587 0.53801653
5                 {bayesian=1} => {class_0_prediction=1} 0.0058257345 0.076411960 0.64186047
```

Figure 16: Aprori applied to a single class.

As as shown in Figure 17, a Sankey Diagram was made to help visualize the results of the association. The lack of thick lines indicate that there is almost no visible association between AI techniques.

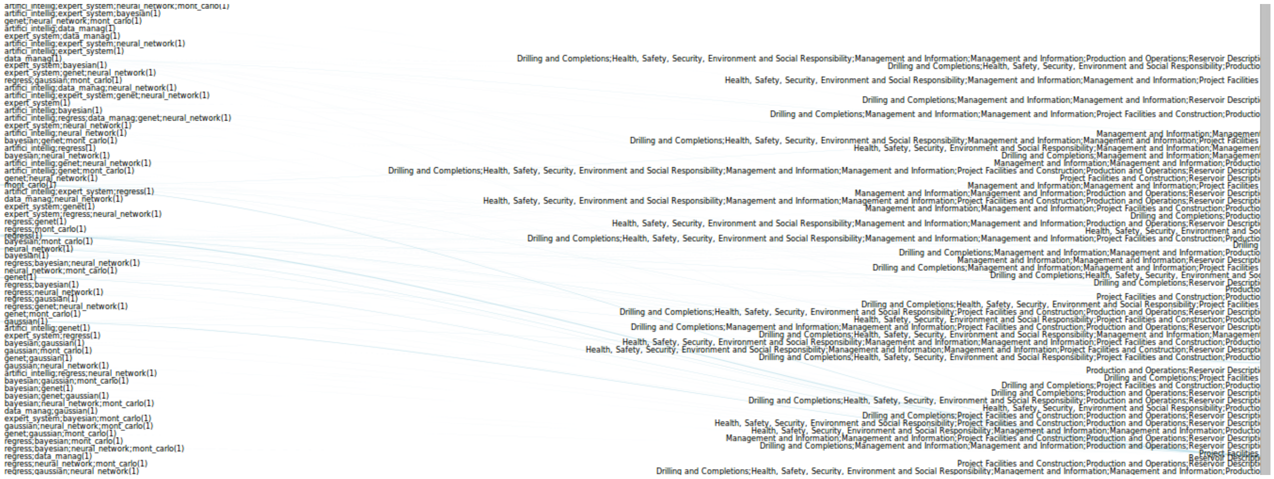


Figure 17: Sankey diagram showing almost no association

The poor results of the association are not due to the lack of association between the AI techniques and the use cases in oil and gas, rather they are due to the lack of good data.[43] The results show that

[43] Many of the papers on OnePetro did not include an abstract.

the frequency of papers which have AI techniques are very small relative to the total number of papers. An alternative method, such as using the full paper corpus that is more likely to contain specific AIPA techniques, should be used to find associations. To validate this assumption, we scraped the full papers which appeared for specific search terms and found much better association results as show in Figure 18.

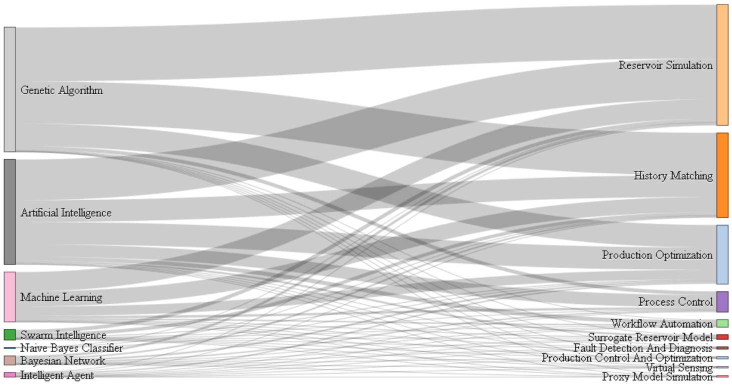


Figure 18: Sankey diagram based on search results.

Conclusions

We found that data mining techniques can be a powerful aid for conducting a literature review. In our initial exploratory data analysis, clustering based on term frequency yielded little insight due to common words across clusters. After reducing the dimensionality of the feature set using inverse document frequencies and Singular Value Decomposition, more meaningful clusters were discovered. Furthermore, a custom vocabulary of common words across clusters was built to aid in classification.

For document classification, the initial Support Vector Machine model with a linear kernel was replaced by a more efficient Stochastic Gradient Descent approach. Random Forests could be built in seconds but consistently were overfitting. Feature engineering was performed by extracting a custom vocabulary from discipline labels. The hand-picked vocabulary performed nearly as well as the full feature set. Performing Singular Value Decomposition to automatically reduce dimensionality performed very poorly since rare but important features were removed. Ultimately, the most optimal results were yielded from n-grams of all words that appear in less than 40% of documents.

A key lesson learned is that accuracy is a misleading metric, especially with an unbalanced mix of positives and negatives in binary classification. Matthew's Correlation Coefficient was the final metric selected as it gave the most complete picture of performance by considering all four quadrants of a Confusion Matrix. This decreased overfitting and improved recall significantly.

Hyperparameter tuning on the Stochastic Gradient Descent classifier demonstrated that elastic net penalty outperforms an L1 penalty on sparse, real world data. Huber loss, which is less sensitive to outliers, performed much better than other loss metrics. The weakness of using log loss while aiming for improved recall may be indicative of low certainty in predictions of positives. A strong presence of false positives in the final model aligns with this notion and it is believed that the manual labeling process may be incomplete.

For the last part of the study, little insight was found by performing Association Rule Mining using Apriori due to the limited occurrence of AIPA techniques in text abstracts. An initial scraping of full papers showed much stronger associations. It is critical that the full text is available in order to obtain an appropriate level of support for the association rules.

Future Work

In the future, this framework for multi-document classification could be formalized and generalized for performing literature reviews for libraries of publications beyond OnePetro.org.

The framework could be improved by considering other metadata such as title, keywords and full paper corpora when performing classification. Additionally, the team identified that further data cleaning could be performed to segregate publications by language. Other classification algorithms are also under exploration. With these improvements, refined results can be obtained to aid petroleum industry engineers and management in understanding the importance of AIPA techniques to oil and gas use cases.

References

Michael J Baker. Writing a literature review. *The Marketing Review*, 1 (2):219–247, 2000.

César E Bravo, Luigi Saputelli, Francklin Rivas, Anna G Pérez, Michael Nickolaou, Georg Zangl, Neil De Guzmán, Shahab Dean Mohaghegh, Gustavo Nunez, et al. State of the art of artificial intelligence and predictive analytics in the e&p industry: A technology survey. *SPE Journal*, 19(04):547–563, 2014.

Kokil Jaidka, Christopher Khoo, and Jin-Cheon Na. *Imitating human literature review writing: an approach to multi-document summarization*. Springer, 2010.

Christopher SG Khoo, Jin-Cheon Na, and Kokil Jaidka. Analysis of the macro-level discourse structure of literature reviews. *Online Information Review*, 35(2):255–271, 2011.

Foltz P. W. & Laham D. Landauer, T. K. Introduction to latent semantic analysis. pages 259–284, 1998.

Joseph Carl Robnett Licklider. Man-computer symbiosis. *Human Factors in Electronics, IRE Transactions on*, (1):4–11, 1960.

SciKitLearn. Gridsearch documentation. a. URL `http://scikit-learn.org/stable/modules/grid_search.html`.

SciKitLearn. Truncatedsvd. b. URL `http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html`.

Ladda Suanmali and Naomie Salim. Literature reviews for multi-document summarization.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.