Unsupervised Machine Learning in Materials Design

Yuyuan Liang

u6319242

A report submitted for the course
COMP8755: Individual project
Supervised by: Nick Birbilis
The Australian National University

June 2020
Except where otherwise indicated, this thesis is my own original work.

Yuyuan Liang
12 June 2020
Acknowledgments

I would like to express my deepest appreciation to my supervisor Nick Birbilis for his guidance, unrelenting support and valuable advisement during this year of the research project. This project provides me with a great opportunity to learn more knowledge about machine learning and data process. I also wish to thank my friends for their help and encouragement when I get trouble. Finally, I would like to express my great gratitude to my parents, who always support me and cover my living expense in Australia.
Abstract

After the second industrial revolution, chromium compounds are widely used as corrosion prevention in all industries. Since their documented carcinogenicity and toxicity, seeking harmless and environment-friendly chromate replacement becomes an urgent task. However, current alternatives from many researches are unable to match the anti-corrosion performance and cost-effectiveness of chromium compounds. Inspiring from the research about capturing latent knowledge from past publications [Tshitoyan et al., 2019], this project applies the similar methods in the study to explore the possibility to find a suitable chromate replacement.

This paper introduces the natural language processing (NLP) algorithm word2vec and evaluates the predictions from the model. Through collecting and classifying numerous abstracts, I implement the word2vec model to predict the possible chromate replacements. The results show the practicability of employing unsupervised machine learning algorithm to seek chromate replacements and the limitations of the model.
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Sigmoid function</td>
<td>4</td>
</tr>
<tr>
<td>2.2</td>
<td>Feedforward neural network architecture [Bengio et al., 2003]</td>
<td>6</td>
</tr>
<tr>
<td>2.3</td>
<td>Continuous bag-of-words and skip-gram models [Mikolov et al., 2013a]</td>
<td>7</td>
</tr>
<tr>
<td>4.1</td>
<td>Top 50 predictions from word2vec model with softmax.</td>
<td>13</td>
</tr>
</tbody>
</table>
List of Tables

4.1 Evaluation of original classifier .................. 11
4.2 Evaluation of tuned classifier ..................... 11
4.3 Top 25 predictions ................................ 12
Chapter 1

Introduction

1.1 Motivation

Chromium compounds have been an indispensable part of modern society as the wide utilization in many industries such as alloy manufacturing, pigments production and wood preservation. Nonetheless, multiple studies point the facts that the exposure of chromium and its derivatives can significantly increase the risk of cancer on any human organ [Khan et al., 2013; Costa and Klein, 2006]. And chromium contamination on water and soil will also transfer the toxicity to plants and aquatic life [Leghouchi et al., 2009; Oliveira, 2012]. Although the related research for substitution chromium compounds has been underway from the 1980s to date, most of the chromate alternatives do not have comparable corrosion protection or cost expectations as hexavalent chromium [Gharbi et al., 2018].

With the rapid advancement of machine learning, there are numerous trials to employ machine learning techniques in various fields. In one study, predictions from the NLP model present the possibility of forecasting the novel thermoelectrics material [Tshitoyan et al., 2019], which implies applying similar methods for discovering new material to substitute the chromate corrosion inhibition is practicable. In our project, we scraped abstracts that focused on chemistry, material science and corrosion, then classified them by logistic regression classifier. After preprocessing the abstracts and extracting chemical formulas, we generated phrases by phraser model of gensim and used them to train a word2vec model. The final target of this experiment is validating the feasibility of predicting novel materials to replace chromium compounds.

1.2 Report Outline

This paper is structured as follows:
Chapter 2: Introduce the background knowledge in this report including logistic regression, word2vec and several related techniques.
Chapter 3: Introduce the Methodology about the data collection and processing, abstract classification and word2vec model training.
Chapter 4: Illustrate the results of classification and word2vec.
Chapter 5: Present the conclusion, the limitations on the current model and what should do in the future.
Background

Considering that we use the logistic regression and NLP model to classify the ab-
stracts and predict, this chapter provides some necessary background materials about
these models.

2.1 Logistic regression

Logistic regression, as an extension of linear regression model, is one of the most
popular machine learning model to solve classification problems. Linear regression
model predicts the continuous target as a weighted sum of predictors, since the
learned relationship between target and feature inputs is linear. However, when it
comes to classification problem, linear regression model can not predict the proba-
bility of each class as its output is the value on the hyperlane.

linear regression equation:

\[ \hat{Y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_i X_i \]

sigmoid function:

\[ h(t) = \frac{1}{1 + e^{-t}} \]

logistic regression equation:

\[ \hat{Y} = \frac{\exp(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_i X_i)}{1 + \exp(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_i X_i)} \]

Instead of fitting a hyperlane, the logistic regression applies the logistic function,
also called sigmoid function, to map continuous output from linear equation into a value between 0 and 1 as shown in figure 2.1.

![Sigmoid function](image.png)

Figure 2.1: Sigmoid function

Compared to the linear regression model, logistic regression does not require the linear relationship between target and predictors because of the non-linear log transformation from linear equation output to probability.

### 2.2 Bag of words(Bow) and Term frequency-inverse document frequency(Tf-idf)

Although logistic regression model can handle the classification problems, it can not directly process the text documents as feature inputs. The text document needs to be converted to a feasible representation as the logistic regression classifier expected input. One prevalent method to extract the features from document is using a bag of words model, which regards document as a bag with its words.

For each term in the corpus, the tf-idf term-weighting scheme, measuring word’s weight by the product of term frequency and inverse document frequency, is used to reflect its importance to a document. The term frequency is the number occurrences of a term in a document. For example, if the word ‘chromium’ occurs five times in an abstract, the term frequency of ‘chromium’ in this abstract is five. Different from the term frequency, document frequency is the number of documents in the dataset that contain the term, and the inverse document frequency is the logarithmically scaled inverse fraction of document frequency. Word has high document frequency means it is a common word across documents like the stopwords ‘the’ and ‘is’. As a result, tf–idf value will increase proportionally with tf value, and idf can reduce the impact of general words that appear frequently in documents on the weight.
2.3 NLP and word2vec

The core knowledge of this project is about NLP, which is an essential subfield of linguistic, computer science and artificial intelligence. With the rise of neural network, the first language model, a one-hidden layer feedforward neural network, was generated in 2001 by [Bengio et al., 2001]. It consists of input, embedding, hidden and softmax layers, and its architecture is shown in figure 2.2. The model takes n previous words to generate word vectors(embeddings) in embedding layer. In comparison with the bag of words model, a sparse vector representation to words, the word embedding represents the words as dense vectors to simplify language modelling. Then the concatenation of word embeddings are fed into hidden layer and applied non-linearity. Finally, the softmax layer produces the probability based on the input of hidden layer. After feedforward neural network, novel neural network models such as recurrent neural networks(RNNs) and long short-term memory networks(LSTMs) were proposed to overcome the limitations of feedforward neural network.

Due to dense values in embedding layer in feedforward neural network model, the computation between embedding layers and the hidden layers is complex. In 2013, the word2vec was firstly proposed by [Mikolov et al., 2013a]. Word2vec models are shallow three-layer neural network model, and it is widely used to produce word embeddings to capture words’ semantic and syntactic similarities and predict word based on its context. For instance, when we use the word ‘chromium’ to predict words in our trained word2vec model, the high similarity results are ‘manganese’ and ‘nickel’, which are the common composition of chromate. In this ex-
ample, ‘chromium’ is closer to ‘manganese’ and ‘nickel’ by cosine distance because they have the similar context in corpus. Though the word2vec model also employs high dimensional word vectors to represent words, the remove of non-linear hidden layer makes the training progress becomes more computationally efficient. This enables the models to train high quality word embeddings on huge dataset with billion words and approximate the certain relations between words, which can not be accomplished on previous models [Mikolov et al., 2013a].

As shown in figure 2.3, two language models have only input, projection and output layers, since the majority of computational complexity comes from the computation between projection layer and hidden layer [Mikolov et al., 2013a]. The continuous bag-of-words (CBOW) model can predict the current words from a window of neighbours words. Similar to bag of words model mentioned above, the orders of words in the history do not influence the prediction. In contrast, the continuous skip-gram Model predicts the surrounding context words based on the current word. And skip-gram model can result in high quality words when the window increases, which indicates skip-gram has high computational complexity but good performance on rare words. In this paper, we focus on the skip-gram model because the idea of ski-gram, that the similar words will appear around similar context words, is applicable in this project.
In the same year, several extensions of skip-gram models are introduced by [Mikolov et al., 2013b]. In this report, we will demonstrate the negative sampling and subsampling as they are applied in our work. Compare to the full softmax in other neural network language models, the hierarchical softmax in word2vec uses a binary tree to represent the output layer and words as leaves. As a result, obtaining the probability distribution only needs to evaluate about log2(W) nodes. Negative sampling, as a substitution of hierarchical softmax, uses k negative samples and the target to train a logistic regression model as a much simpler method to distinguish the target. This way only needs to modify a small part of the weights, not all weights for each training sample. In addition, another extension, subsampling, can balance the infrequent and frequent words such as ‘the’, ‘is’, ‘are’ etc. After setting a threshold t, the subsampling will remove some instances of the words whose frequency is greater than t while keeping their rank of frequency. It is evidenced that subsampling of frequent words can increase the training speed and lead to a higher accuracy to infrequent words [Mikolov et al., 2013b].
Methodology

3.1 Data collection and processing

In the stage of data collection, we obtained around 1.2 million abstracts come from the field of chemistry, material science and corrosion by the Springer Nature application programming interface(API), Elsevier’s Scopus and Science Direct API. For the data processing part, we removed the empty abstracts firstly. And secondly, abstracts whose genres are ‘Announcement’, ‘BookReview’, ‘Erratum’, ‘EditorialNotes’, ‘News | Events’ and ‘Acknowledgments’ were eliminated from the dataset by regular expression replacing. The same method was also applied on abstracts with title including ‘Foreword’, ‘Prelude’, ‘Commentary’, ‘Workshop’, ‘Conference’, ‘Symposium’, ‘Comment’, ‘Retract’, ‘Correction’, ‘Erratum’, ‘Memorial’ and ‘Acknowledgement’.

Except for certain types of abstract, we also used the langdetect Python library and text search to remove the non-English abstracts and the abstracts that consist of non-English parts. Thirdly, copyright information, control characters in ASCII and leading words such as ‘Abstract’ in some abstracts were also removed through regular expression match. We also used similar methods on Elsevier data to eliminate Elsevier tag such as ‘<ce. >’ and ‘<inf/ >’. During the data preprocessing stage, we found that some abstracts have no space between the dot and words. This issue would cause the tokenize function identifies some words as one word, and all sentences in abstracts would merge into one sentence. We used a general string match method to find the dot and sentence the abstracts correctly. Finally, we eliminated the duplicate abstracts by Pandas function in Python. After data cleaning process, there are approximately 750,000 abstracts in the dataset.

3.2 Abstract classification

For the purpose of mining chromate replacement, appropriate corpus selection is significant for the prediction accuracy. As reported in the paper of [Tshitoyan et al., 2019], using relevant abstracts as training corpus can decrease the noises in embed-
Methodology

...dings and lead a better performance on the language model. Therefore, we abandoned the abstracts that unrelated to corrosion, alloy and inorganic material science except for abstracts about the organic corrosion inhibitors.

To train the classifier, we labelled 1100 abstracts that randomly selected from corrosion and other research areas as ‘0’ and ‘1’ corresponding ‘irrelevant’ and ‘relevant’. In the labelled abstracts, 582 of them were labelled as ‘0’, and 518 of them were labelled as ‘1’. The selection of abstracts is not completely random because the number of relevant abstracts in a randomly selected set is extremely less than irrelevant abstracts’. During the preprocessing stage, the punctuations and stopwords tokens were removed after tokenizing the sentences, and each abstract was represented as bow vector with words’ tf-idf values. Eventually, we used the linear logistic regression classifier from sklearn Python library with five-fold cross-validation to classify the abstracts.

3.3 Training word2vec model

Before training the word2vec model, the abstracts were tokenized by ChemDataExtractor. When the chemical formulas were detected by pymatgen and regular expression, we saved them in a formula file and normalised them to remove the ambiguity from the different formula writing style. Besides, the valence state and unit in tokens were also split from the original token. And numbers exclude the ones in the formula were convert to a ‘nUm’ string, which can reduce the vocabulary size. Finally, all the tokens except for chemical formulas and elements were changed to lowercase and removed accents.

Following preprocess the abstracts, we used the phraser model and the word2vec model in gensim to generate phrases and train. For the phraser model, we used the phrase depth of 4, phrase count of 10 and threshold of 15. The depth parameter means that the phraser will repeat the phrase function 4 times. We also used the common terms to generate long token phrases, and exclude terms to remove the number and punctuation. For the word2vec model, we used dimensions of 200, window length of 8, initial learning rate of 0.01, minimal learning rate of 0.001, minimal count of 5, subsample of 0.0001. And we used negative of 15 to pursue a better performance on skip-gram model [Mikolov et al., 2013b]. As mentioned in chapter 2, we used the skip-gram model to predict the chromate replacement because compounds similar to chromate may have identical context.
Chapter 4

Results

4.1 Classification results

The results of abstract classification are shown in table 4.1, where the classifier can reach a high accuracy of 93%, where prediction and recall are all 93%.

<table>
<thead>
<tr>
<th></th>
<th>precision</th>
<th>recall</th>
<th>F1 score</th>
</tr>
</thead>
<tbody>
<tr>
<td>not relevant</td>
<td>0.96</td>
<td>0.82</td>
<td>0.88</td>
</tr>
<tr>
<td>relevant</td>
<td>0.84</td>
<td>0.96</td>
<td>0.90</td>
</tr>
</tbody>
</table>

Table 4.1: Evaluation of original classifier.

The precision is the ratio to measure how many abstracts were classified correctly for each class. The recall is the ratio to measure how many relevant abstracts were classified as ‘relevant’. And the F1 score is the harmonic mean of precision and recall. Since we need to include as more relevant abstracts as possible in our corpus, we turned low the decision baseline to reach high recall. The results of modified abstract classifier are shown in table 4.2. The classifier reaches a high recall of 96%, which indicates 96% of relevant abstracts were allocated to our corpus. After abstract classification, we obtained 272,000 relevant abstracts.

<table>
<thead>
<tr>
<th></th>
<th>precision</th>
<th>recall</th>
<th>F1 score</th>
</tr>
</thead>
<tbody>
<tr>
<td>not relevant</td>
<td>0.93</td>
<td>0.93</td>
<td>0.93</td>
</tr>
<tr>
<td>relevant</td>
<td>0.93</td>
<td>0.93</td>
<td>0.93</td>
</tr>
</tbody>
</table>

Table 4.2: Evaluation of tuned classifier.

4.2 Word2vec prediction results

We trained the skip-gram model with hierarchical softmax and negative sampling respectively. And we used the ‘chromate’ and ‘replacement’ as key words to make
results of these two models are presented in the table 4.3.

<table>
<thead>
<tr>
<th>Predictions(softmax)</th>
<th>Similarity</th>
<th>Predictions(negative sampling)</th>
<th>Similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bi13Cr5</td>
<td>0.6413889527320862</td>
<td>Bi13Cr5</td>
<td>0.6170487403869629</td>
</tr>
<tr>
<td>INi7Hf2</td>
<td>0.5784975290298462</td>
<td>K2Cr2O4</td>
<td>0.5960352420806885</td>
</tr>
<tr>
<td>C2O42NO3</td>
<td>0.5585286617279053</td>
<td>CrNiMoCu</td>
<td>0.5462764501571655</td>
</tr>
<tr>
<td>K2Cr2O4</td>
<td>0.5543444156646729</td>
<td>Cu38Zn</td>
<td>0.5442754626274109</td>
</tr>
<tr>
<td>NiO42</td>
<td>0.553759574801367</td>
<td>K2Ni(CN)4</td>
<td>0.538384838867188</td>
</tr>
<tr>
<td>CrNiMoCu</td>
<td>0.5451786518096924</td>
<td>NiO42</td>
<td>0.537669837474823</td>
</tr>
<tr>
<td>ZnSO3</td>
<td>0.5429255962371826</td>
<td>INi7Hf2</td>
<td>0.5360739231109619</td>
</tr>
<tr>
<td>Co(NH3)3</td>
<td>0.5381604433059692</td>
<td>LiCILiFGdF3</td>
<td>0.535133886535645</td>
</tr>
<tr>
<td>(OH)PO4</td>
<td>0.5224480628967285</td>
<td>Cu10Al10Zn</td>
<td>0.5341971516609192</td>
</tr>
<tr>
<td>NaFe3(SO4)2(OH)6</td>
<td>0.5202199816703796</td>
<td>HgCO3</td>
<td>0.5339683890342712</td>
</tr>
</tbody>
</table>

Table 4.3: Top 25 predictions.

As we can see from table 4.3, the predictions still contain the chromium compounds and chromium alloy, since they have similar properties to chromate or only have similar context words to chromate. For example, the reason why the Bi13Cr5 has the highest similarity to chromate replacement is its context words in the paper in 2017 [Mikhaylovskaya et al., 2017] are ‘substituted’, ‘chromium’ and ‘chromate’, which are highly similar to the ‘chromate’ and ‘replacement’. Another example is K2Cr2O4, which presents great capability against corrosion. Besides, some predictions may be the related products of chromate or connect to chromate by other words(context). ZnSO3, as a good example, is corrosive and one of the products of corrosion from galvanized steel [Yan et al., 2013]. It may connect to chromate by ‘corrosion’ because its properties are unconnected to chromate.

To visualize the similarity of the results, we applied dimension reduction to the word embeddings and plotted them in a 2-dimension figure. The top 50 predictions were also annotated in the figure.

As shown in figure 4.1, the distances between the predictions can be regarded as the similarities. Most of the predictions are concentrated in upper right areas, and some positions of them almost coincide, which implies they have extremely high similarity to each other.
Figure 4.1: Top 50 predictions from word2vec model with softmax.
Conclusion

In this work, we collected 1.2 million abstracts and categorized them through a logistic regression classifier. These abstracts were converted to high dimensions vectors in word2vec model and used for predicting the similar compounds of chromate. Through the analysis of predictions from the skip-gram model, we conclude that it is feasible to find the compounds having similar properties to chromate. And the predictions illustrate the possibility of predicting new materials to replace the chromate.

5.1 Limitation

We noticed that there are limitations in our project. First of all, the general output of word2vec models has semantic and syntactic relationships, that is, the output is effortless to validate. However, the validation of predictions from our model requires material science professionals. Secondly, as the main idea of the language model is extracting the knowledge from past publication, predicting a chemical compound never appeared before is impossible. Finally, the common limitation on word2vec model is there is no effective method to evaluate the model, especially for our project. Most of the word2vec models require modification during the application in reality. For our model, the time to evaluate the actual performance maybe a year or more.

5.2 Future Work

In the future, this research can be extended into two parts. In the first part, we are exploring the performance of prediction on other advanced models. One of the valuable alternatives is pretrained model. The pretrained language models only required unlabelled data and enable to learn from a small dataset, which is much suitable for this project. On the other hand, the current work can be optimised in the preprocessing methods and extended to many other research areas.
Bibliography


INDEPENDENT STUDY CONTRACT
PROJECTS

Note: Enrolment is subject to approval by the course convenor

SECTION A (Students and Supervisors)

UniID: __u6319242________
SURNAME: __Yuyuan________ FIRST NAMES: __Liang________
PROJECT SUPERVISOR (may be external): __Nick Birbilis________
FORMAL SUPERVISOR (if different, must be an RSSCS academic):

COURSE CODE, TITLE AND UNITS: COMP8755 Individual Computing Project 12 units

COMMENCING SEMESTER ☐ S1 ☒ S2 YEAR: _2019_ Two-semester project (12u courses only): ☒

PROJECT TITLE:
Machine learning in materials design

LEARNING OBJECTIVES:
1. Learning via data mining scientific literature
2. Experience with unsupervised word embedding training for material science
3. Experience with using word embeddings to predict novel materials
4. Specifically focus on chrome free inhibitors for corrosion protection (a major international challenge)

PROJECT DESCRIPTION:
Collecting and processing data from materials science literature. This is possible using online databases accessible via the ANU.
Using API to obtain related topic abstracts and processing them by text search and regular expression matching.
Using a linear binary classifier based on logistic regression to label the abstracts as ‘relevant’ or ‘not relevant’.
Tokenizing the relevant abstract to produce individual words for text corpus.
Preprocessing the words correctly (etc. selective lower-casing and deaccenting) to improve the training results.
Constructing word embeddings by training Word2vec which use the information about the co-occurrences of words in text corpus.
Optimising the hyperparameters for performance on grammatical and materials science analogies.
Testing and predicting novel materials by word embeddings and embedded relationships.
Writing project report.
Rationalisation of learning and findings with subject matter experts.

Research School of Computer Science

Form updated Jun 2018
ASSESSMENT (as per the project course’s rules web page, with any differences noted below).

<table>
<thead>
<tr>
<th>Assessed project components:</th>
<th>% of mark</th>
<th>Due date</th>
<th>Evaluated by:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Report: style: ______________ (e.g. research report, software description...)</td>
<td>(min 45, def 60)</td>
<td>(examiner)</td>
<td></td>
</tr>
<tr>
<td>Artefact: kind: ______________ (e.g. software, user interface, robot...)</td>
<td>(max 45, def 30)</td>
<td>(supervisor)</td>
<td></td>
</tr>
<tr>
<td>Presentation:</td>
<td>(10)</td>
<td>(course convenor)</td>
<td></td>
</tr>
</tbody>
</table>

MEETING DATES (IF KNOWN):

STUDENT DECLARATION: I agree to fulfil the above defined contract:

Yuyuan Liang

Signature Date

SECTION B (Supervisor):

I am willing to supervise and support this project. I have checked the student's academic record and believe this student can complete the project. I nominate the following examiner, and have obtained their consent to review the report (via signature below or attached email)

N. Roberts

5/8/19

Signature Date

Examiner:
Name: Hanna Suominen

Signature

(Nominated examiners may be subject to change on request by the supervisor or course convenor)

REQUIRED DEPARTMENT RESOURCES:

SECTION C (Course convenor approval)

Research School of Computer Science
Signature  Date