A Billion-Dollar Material
Finding Ultra-Low Emittance Photocathode Materials for Linear Accelerators

Michael Cai
Department of CS
Stanford University
mcai88@stanford.edu

Michael Cooper
Department of CS
Stanford University
coopermj@stanford.edu

Jennie Yang
Department of MSE
Stanford University
jenniey@stanford.edu

Abstract

We investigate AI-based approaches for improving computation times of the thermal emittance of photocathode material candidates. We frame the problem as a binary classification problem, describe three standardized approaches for representing material features as inputs for machine learning models, and evaluate the performance of common machine learning models on this task. We conclude that a neural network model based on the average atomic properties for each material is the best performing model, though more advanced data representations could be applied in future to increase prediction accuracy.

1. Introduction

The Stanford Linear Accelerator Center **(SLAC)** is one of the world's premier tools for studying advanced physics and chemistry. In this project, we explore the application of machine learning to the field of Computational Materials Science to find new materials to satisfy a set of desired properties for SLAC experiments.

A team headed by Associate Professor **Dr. Evan Reed** of Stanford's Department of Materials Science and Engineering is in the early stages of a project in collaboration with SLAC, the goal of which is to use computational methods to predict materials with sufficiently low **thermal emittance** to use in advanced photocathodes in particle accelerators. Thermal emittance is inversely proportional to beam brightness, so low-emittance materials enable sensing with the use of more powerful electron beams and, therefore, new experiments.

The Reed Group currently deploys a model based on Density Functional Theory **(DFT)**, an accurate but computationally expensive method for generating a material's electronic band structure, from which thermal emittance can be easily computed. The DFT model takes anywhere between several hours to a full day to predict the emittance of one candidate material; thus, the goal of our project is to use machine learning on these DFT results in hopes of significantly expediting the prediction process.

2. Related Work

In 2010, Dowell *et al* proposed a three-part plan of photocathode R&D: (1) study cathode formation and emission with existing diagnostic strategies, (2) model emission and electron dynamics in cathodes, and (3) test cathodes in operation. [1] This plan is consistent with the typical process of materials research, which centers on studying known (though perhaps poorly-understood) materials and developing physical models to describe them. For example, in accordance with step (1) of this R&D plan, Bazarov *et al* used a solenoid scan to measure the emittance of a cesium potassium antimonide photocathode. [2]

While such physical experiments are vital to materials research, they are time-consuming and perform poorly at scale. These techniques are impractical for screening a large quantity of materials or analyzing materials which cannot yet be fabricated. Computational models, on the other hand, allow researchers to study the properties of materials without requiring physical samples. In 2015, Li *et al* explored the connection between the electronic band structure of photocathodes and the momentum of the emitted electrons [3], which helped make it possible to determine thermal emittance through DFT. While such computation is faster than physical experimentation, it can still prove too slow for the study of a large quantity of materials. Thus, machine learning may prove a faster early-screening tool to complement computational materials research.

Machine learning in materials science is a new frontier: machine learning-compatible representations of materials is very much an active area of research. For example, in April 2019, Himanen *et al* released DScribe, a software package for generating several different kinds of ML-friendly descriptors of materials, including the Coulomb and Ewald sum matrices. [4] Unfortunately for our purposes, the representations provided by DScribe are best applied when dealing with materials all of the same class; our dataset is too

diverse to appropriately apply the DScribe library.

Instead, one of the material-representation methods we use in our investigation is a variant of the "elemental descriptors" method described by Cubuk, *et al*, which is based on the composition of the material rather than its structure [5], making it suitable to describe the wide range of materials in our dataset.

3. Data and Features

3.1. Data

The Reed Lab provided us a list of candidate materials. Of these candidates, 9,436 also have associated emittance data, calculated through a DFT-based approach. Each material's emittance data is represented as a series of 100 emittance values as a function of incident photon energy, starting at the material's band gap energy (E_g) and going up to $E_g + 5$ eV. For each material, we extracted our features via the Materials Project API (documented here) and through the `mendeleev` Python package (documented here) [7].

3.2. Input Representations

Representing the materials in our dataset was a significant challenge, as no known representation can fully capture both the physical and structural properties of a material, and work well with both molecular and lattice-based materials. In this project, we evaluated performance on three different data representations.

3.2.1 Unit Cell Representation

Our preliminary method of representing each material features a **71-element vector** encoding the unit cell (smallest repeating unit for a crystal). The first three entries of the vector give the lattice parameters, which are the dimensions of its unit cell (a, b, c), the next three entries are the alpha-beta-gamma angles between the a,b,c axes (α, β, γ), and the seventh entry gives the relaxed unit cell volume (v). See Figure 1 for a visual representation of these parameters. The final 64 entries give the atomic number of each of the first 16 atoms (e_i) in the unit cell followed by the locations of each atom in the material as a proportion of each a-b-c dimension (a_i, b_i, c_i).

$$\vec{X} = [a, b, c, \alpha, \beta, \gamma, v, e_0, a_0, b_0, c_0, ..., e_{15}, a_{15}, b_{15}, c_{15}]$$

For non-crystalline materials, the unit cell is set such that it contains one molecule of the material. All of our unit cell information for both crystalline and non-crystalline materials is drawn from the Materials Project's database. Unused entries in each feature vector are zero-padded to standardize input length.

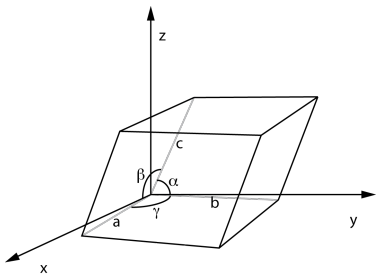

Figure 1: Visual depiction of a unit cell. The coordinates of the atoms within the unit cell are given relative to the origin and the a, b, c axes.

3.2.2 Average Properties Representation

Our next representation uses a **16-element vector**, where each component is the mean of some atomic property over all of the atoms in that material, based on its reduced chemical formula. Our vector contained the following atomic properties:

Atomic volume	Atomic weight	Boiling point
Covalent radius	Density	Electron affinity
Electronegativity	Period table group	Heat of formation
Heat of fusion	Heat of evaporation	Lattice constant
Melting point	Period	Polarizability
Specific Heat at $T = 20°C$		

Properties of atoms without a value for that property were given values of 0. Properties of materials without a value for that property - if, for example, all atoms in the material lack that property - were assigned a value equal to the average property value across all materials. We utilized this method to prevent such datapoints from skewing our classifier in an undesirable way.

3.2.3 Combined Representation

Our third representation is the concatenation of the unit-cell vector and the average properties vector, resulting in am **87-element vector** for each material.

3.3. Emittance Labels

For each material, we defined its minimum emittance as the lowest *nonzero* emittance value (rounded to the nearest tenth to account for the ± 0.05 error from the calculations) achieved across at least two consecutive incident energies. Finally, we assigned a score of **one** to examples with minimum emittance **less than or equal to 0.2**, and a score of **zero** to those **greater than 0.2**. Figure 3 illustrates the process by which labels were determined.

Our labelled dataset contained **3276** positive examples and **6160** negative examples for a total of 9436 materials. We randomly shuffled the data and performed a **80/10/10** train/validation/test split. The training set contained **2577**

positive and **4971** negative examples.

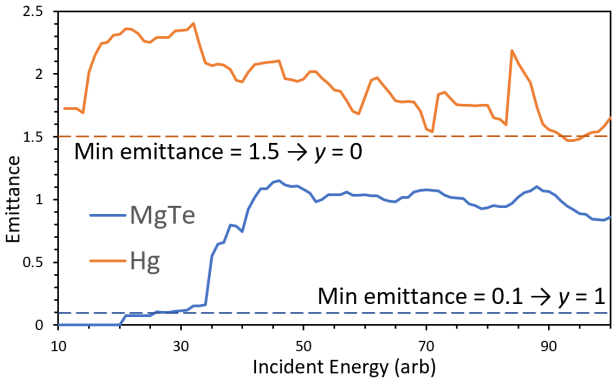

Figure 2: Example visualization of emittance data for Magnesium Telluride and Mercury and their resulting labels, $y = 0$ and $y = 1$, respectively.

4. Methodology

Our project evaluated a number of different machine learning techniques, including both traditional and deep learning models, to determine the best model for this classification task. Each model was tuned on each of our different data representations. This section describes each model, as well as the techniques used for model evaluation.

4.1. Traditional ML Techniques

As part of our investigation of this data, we employed several traditional machine learning techniques. We implemented each classifier using its sklearn module. The models and hyperparameters deployed are as follows:

- Logistic Regression - L_2 norm, 10^7 iterations max
- SVM - penalty of 1.0, RBF kernel, kernel coefficient of $1/\text{num features}$
- Naive Bayes - Bernoulli model, no Laplace smoothing
- Random Forest - 100 trees, Gini impurity criterion
- K-Nearest Neighbors - $k = 2$, neighbors weighted by inverse of distance

4.2. Neural Network

In addition to the traditional ML models described above, we experimented with different fully-connected neural network models, adjusting hyperparameters such as the learning rate, number of epochs, number of hidden layers, and layer sizes. The neural network implementation we used is the sklearn Multilayer Perceptron Classifier class. Our best model across any data representation contains **7** hidden layers of sizes **(64, 128, 256, 512, 256, 128, 64)**, each followed by a **ReLU** nonlinearity. We used a learning rate of **0.0008**, the **binary cross entropy** (BCE) loss function, an L2 regularization parameter of **0.0001**, and an **Adam** optimizer to train the model.

4.3. Evaluation Metrics

Due to the asymmetry in our positive and negative examples, we knew that a simple accuracy metric would be insufficient to evaluate the performance of our model. We employed the following four metrics to evaluate model performance: **accuracy** (acc), **precision** (pr), **recall** (re), and **F1 score** ($f1$). Given the asymmetric label frequency in our dataset, the the $f1$ score is a better balanced metric than straight acc to evaluate the model both on pr and re. Our metrics are calculated using the following formulae:

$$acc = \frac{\text{True Pos + True Neg}}{\text{True Pos + True Neg + False Pos + False Neg}}$$

$$pr = \frac{\text{True Pos}}{\text{True Pos + False Pos}}$$

$$re = \frac{\text{True Pos}}{\text{True Pos + False Neg}}$$

$$f1 = \frac{2 * pr * re}{pr + re}$$

5. Results and Discussion

5.1. Results of Learning Methods on Validation Set

5.1.1 Unit Cell Representation

Method	***acc***	***pr***	***re***	***f1***
LogReg	0.626	0.064	0.106	0.080
SVM	0.674	0.105	0.188	0.135
NB	0.552	0.230	0.336	0.273
RF	0.733	0.218	0.423	0.287
kNN	0.686	0.302	0.549	0.390
NN	0.668	0.331	0.585	0.423

5.1.2 Average Properties Representation

Method	***acc***	***pr***	***re***	***f1***
LogReg	0.639	0.081	0.137	0.102
SVM	0.685	0.130	0.235	0.167
NB	0.622	0.0	0.0	N/A
RF	0.749	0.273	0.541	0.363
kNN	0.676	0.268	0.479	0.344
NN	0.748	0.333	0.658	0.442

5.1.3 Combined Representation

Method	*acc*	*pr*	*re*	$f1$
LogReg	0.667	0.162	0.286	0.207
SVM	0.642	0.036	0.062	0.046
NB	0.551	0.231	0.336	0.274
RF	0.758	0.237	0.476	0.317
kNN	0.708	0.302	0.566	0.394
NN	0.734	0.330	0.641	0.436

5.2. Comparison of Learning Methods

For each method, the neural net largely outperforms other methods, especially with the Average Properties and Combined representations. While the random forest achieved slightly higher accuracy across representations, the neural net was significantly superior in the other three metrics. Notably, the neural network achieved a higher $f1$ score, which we explained above provides a better measure of performance given the nature of our dataset. On the test set, the neural net with Average Properties gave the following results:

Model	*acc*	*pr*	*re*	$f1$
NN w/ AvgPropRep	0.701	0.296	0.573	0.390

5.3. Bias and Variance Analysis

To analyze the efficacy of our neural net, we plot our training and cross-validation accuracies against number of training examples. We find that our model achieves greater than 90% accuracy on the training dataset and much lower accuracy in the cross-validation, which shows that our model overfits our dataset and that regularization may improve the neural network model. We regularized our model with $\mathcal{L}_2$ regularization, testing weights between 0.01 and 0.9, but regularization did not significantly improve model performance, nor did it have much of an impact on the training accuracy, which suggests that regularization is not enough to combat the overfitting occuring in our model.

5.4. Discussion

Our work demonstrates that it is possible to classify emittances with a greater than 70% degree of accuracy using machine learning. We demonstrate that classification via machine learning is significantly faster than the DFT method: while DFT takes hours to days per material, our neural network predicts the emittance classification of one material in less than a second. The neural network model that we demonstrate, though, is an imperfect substitute for DFT; while DFT is able to calculate the exact emittance every time, we obtain a test set $f1$ score of 0.39. The more rapid classification provided by our system, though, provides a strong starting point for materials science research: researchers could, perhaps, employ our system to narrow a broad field of candidate materials, then run DFT on the most promising candidates.

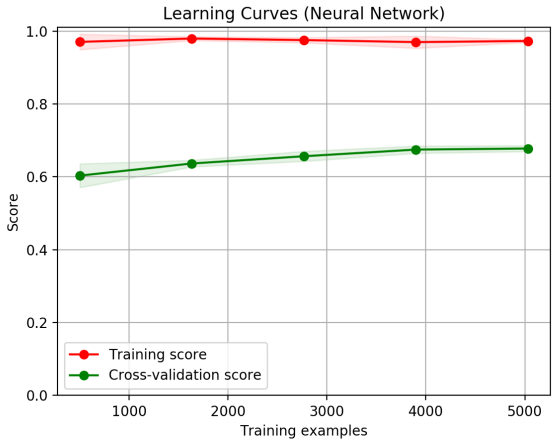

Figure 3: Training and cross-validation learning curves for our seven-layer neural net with Average Properties representation (unregularized).

We hypothesize several reasons as to why our system is unable to obtain a level of performance to rival DFT. First, observe the complex, multistep nature of the learning problem: we are asking the machine learning model to do the equivalent of generate the emittance vs. incident energy graph (Fig. 2), then minimize the graph. Generating the graph is a significant challenge in itself: the electronic band structure, and thus the emittance graph, can vary substantially based on minor changes to the input features. Second, perhaps a model with more data would perform better. The cross-validation curve (Fig. 3) increases over with the number of training examples, which suggest that adding more data would improve performance. We conjecture that our 9,436-point dataset, though substantial, may be too small for our neural net to comprehensively establish the impact of the value of each feature on the overall emittance classification (this may be why our neural network performed best on the average properties representation, which contained the fewest features). Thirdly, there is room for improvement in our data representation; below, we compare our material representation models one-by-one.

5.5. Discussion of Representation Methods

5.5.1 Unit Cell Representation

The main strength of the unit cell representation is that it encodes the structure of the material. The unit cell representation is an encoding of the entire input of DFT calculations. Theoretically, it thus encodes all of the information needed to predict emittance. However, the unit cell representation suffers from inconsistent vector lengths, as not all materi-

als will have the same number of atoms in their unit cells. To address this, we truncated all vectors to represent up to 16 atoms and padded with zeros wherever needed; this may have brought about unintended consequences in our training.

Another shortcoming of the unit cell representation is that each material does *not* have many different encodings. In particular, the feature vector could have its component atoms reordered and still represent the same material, or the origin of the unit cell could be made to start at a different position relative to its atoms, such that all of the coordinates end up being translated by some amount; such a change would still result in the same material. A third drawback of the unit cell representation is that it fails to successfully capture the relationships between atoms in the material. By merely listing the identities and positions of the atoms, the unit cell representation does not capture which atoms are bonded to each other, the quantity and arrangement of each atom's nearest neighbors, the identities of those nearest neighbors, among others.

5.5.2 Average Properties Representation

The main strength of this representation is that a feature vector can be generated for any material, as long as its chemical composition is known. Thus, even materials that have never been synthesized can be easily represented. The use of a fixed list of properties also more successfully standardizes the length of each vector. However, due to the wide variety of compounds in our dataset, not all information is available for all elements in `mendeleev`, so this representation still requires padding to some degree. Furthermore, many different materials can have the same composition, which would result in identical feature vectors.

Perhaps the main weakness of the average properties representation is that it loses explicit structural/ crystallographic information about the material, which means the link between materials properties and emittances, which are derived from the output of DFT calculations, is quite indirect. Perhaps this explains why the neural net outperformed all of our other learning methods, as neural network architecture can somewhat compensate for the oblique link between our features and labels. It is possible that adding more properties to our feature vectors would improve the performance of this method.

Another potential avenue of improvement would be to include properties of the whole material, not just its atoms. This would allow the curation of features that, through physical or empirical models, are thought to correlate strongly with emittance, rather than just a wide variety of less-related atomic properties. However, there is still room for research into finding such features, and it is not guaranteed that such properties are already known for all materials being screened; this is especially true for materials for which we can not yet fabricate physical samples.

6. Conclusions and Future Work

In conclusion, our work shows that there is significant promise for machine learning to add valuable insight as a rapid-screening mechanism for materials science research.

The design of molecular representations for computational materials science is an active area of research, and future work may be able to build off of novel representations to achieve better performance. Promising representations for future study include the models implemented in `DScribe`, provided the input data is stratified to use the appropriate representation, as well as the MEGNet model described by Chen *et al*, in which both molecules and crystals are encoded as graph networks [6].

In future work, decomposing our multistep classification problem into discrete steps, each modeled as a separate machine learning task, may prove fruitful. For example, machine learning may be applied to determine emission spectra (after which standard techniques are used to minimize the curve), or used to generate the atomic band structure (thereby supplanting DFT). The latter, though a lofty task, would be a significant step forward in the field of computational machine learning.

7. Appendix

Our Github repository can be found at `https://github.com/theCaiGuy/Photocathode-Materials`

8. Acknowledgements

We would like to thank the Reed group both for inspiring this project and for being an incredible resource for us throughout. We would especially like to acknowledge Evan Antoniuk, a PhD student in the Reed group, who has been our primary contact throughout this process and provided us with both the dataset to make the project happen, and the knowledge to understand the task at hand.

Features for each material were extracted from the open-source Materials Project API and the `mendeleev` package. All model implementations were drawn from the open-source SciKit Learn package.

9. Contributions

Jennie Yang and Michael Cai worked on extracting and cleaning data for each material from an online database. Jennie also took care of the literature review and other areas of the project that required materials science background.

Michael Cooper and Jennie Yang built the traditional machine learning models (log reg, NB, SVM, random for-

est), while Michael Cai built and fine-tuned the neural network. All group members contributed to running the models across different representations, compiling performance data, and debugging issues with our implementations and dataset.

All three group members contributed equally to the poster and project writeup.

References

[1] D. H. Dowell *et al*, Cathode R&D for Future Light Sources, *Nuclear Instruments and Methods in Physics Research Section A*, vol. 622, no. 3, pp. 685697, Oct. 2010. 1

[2] I. Bazarov *et al*, "Thermal emittance measurements of a cesium potassium antimonide photocathode," *Applied Physics Letters*, vol. 98, 02 Jun. 2011. 1

[3] T. Li *et al*, "Emission properties of body-centered cubic elemental metal photocathodes," *Journal of Applied Physics*, vol. 117, 10 Mar. 2015. 1

[4] L. Himanen *et al*, "DScribe: Library of Descriptors for Machine Learning in Materials Science," `arXiv:1904.08875 [cond-mat.mtrl-sci]`, 18, Apr. 2019. 1

[5] E. Cubuk, *et al* "Screening billions of candidates for solid lithium-ion conductors: A transfer learning approach for small data" *Journal of Chemical Physics*, vol 150, 3 Jun. 2019. 2

[6] C. Chen *et al*, "Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals," `arXiv:1812.05055v2 [cond-mat.mtrl-sci]`, 28 Feb. 2019. 5

[7] L. M. Mentel, `mendeleev` - A Python resource for properties of chemical elements, ions and isotopes, 2014. 2