Representing Multiword Chemical Terms through Phrase-Level Preprocessing and Word Embedding

Liyuan Huang and Chen Ling*

Toyota Research Institute of North America, 1555 Woodridge Avenue, Ann Arbor, Michigan 48105, United States

Supporting Information

ABSTRACT: In recent years, data-driven methods and artificial intelligence have been widely used in chemoinformatic and material informatics domains, for which the success is critically determined by the availability of training data with good quality and large quantity. A potential approach to break this bottleneck is by leveraging the chemical literature such as papers and patents as alternative data resources to high throughput experiments and simulation. Compared to other domains where natural language processing techniques have established successes, the chemical literature contains a large portion of phrases of multiple words that create additional challenges for accurate identification and representation. Here, we introduce a chemistry domain suitable approach to identify multiword chemical terms and train word representations at the phrase level. Through a series of special-designed experiments, we demonstrate that our multiword identifying and representing method effectively and accurately identifies multiword chemical terms from 119,166 chemical patents and is more robust and precise to preserve the semantic meaning of chemical phrases compared to the conventional approach, which represents constituent single words first and combine them afterward. Because the accurate representation of chemical terms is the first and essential step to provide learning features for downstream natural language processing tasks, our results pave the road to utilize the large volume of chemical literature in future data-driven studies.

1. INTRODUCTION

Artificial intelligence (AI) has demonstrated plenty of remarkable progresses in the recent years, from game playing strategies AlphaGo1 to accurately classify skin cancer2 and from image classification3 to neural machine translation system.4 Material science has also been greatly propelled by these data-driven-based AI technologies.5 For example, machine learning and deep learning techniques were used for predicting target properties of chemicals,6−14 in chemical discovery pipelines15−17 and in reaction prediction.18,19 Among all of these applications, the foundations of AI models and algorithms are to learn underlaying statistical relationships and patterns in the training data.20 In material science, AI models can be trained on data generated through “high-throughput” simulation and/or experiments. For example, several publicly accessible repositories are available, such as Materials Project,21 Open Quantum Materials Database (OQMD),22 and AFLOW software framework,23 to supply materials data from high-throughput density functional theory calculation in a large quantity and well-controlled quality. However, due to the computational cost, they are typically limited to a few fundamental properties such as formation energy and band gap values. Materials data can also be obtained by consolidating different handbooks. This way usually generates a set of a few hundred examples.24 An alternative to acquire material data is to extract information from existing material literature such as patents and publications by text mining. For example, Kim et al. mined the synthesis parameters of oxide materials from 640 000 journal articles.25 Based on the extracted information, they succeeded in predicting the important parameters required to inorganic materials leveraging machine learning techniques.26−30 Elton et al. extracted information on the properties and functionalities of energetic materials based on a total of 3136 patents.31 More recently, Tshitoyan et al. conducted an unsupervised word embedding based on a total of 1.5 million abstracts from materials science, physics, and chemistry publications to capture complex materials science concepts and successfully applied the learned knowledge to recommend new thermoelectric materials.32 Before widely implemented in material science, text mining had been successfully and frequently employed in other domains, such as linguistic, biomedical, and marketing areas.33−36 Compared to the general literature used by these studies, chemical literature is unique in terms of the abundance of chemistry-specific linguistic lexicons. Townsend et al. showed that at least 14 kinds of chemistry-specific lexicons were necessary to sufficiently describe chemical concepts in organic chemistry.37 It becomes more concerning because the
majority of these terms are composed of multiple words. For example, most chemical names are composed of more than one single word, such as “lithium chloride” and “lithium hydroxide monohydrate”. Reactions such as “double replacement” and “oxygen evolution reaction” and apparatus such as “graduated cylinders” are also examples of this type. The frequent appearance of these multiword phrases also creates great challenges to appropriately and accurately embed the semantic meanings in vectors, which serve as key features in a variety of downstream tasks, such as part-of-speech tagging, sentiment analysis, and text summarization. Currently, the most frequently used technique for the phrase identification is named entity recognition, which leverages linguistic grammar-based techniques or statistical models, while for the word representation, the conventional method includes two steps: first, represent each single constituent word independently and then combine them up to a multiword phrase representation afterward. For example, to represent the phrase “lithium chloride,” one needs to represent the individual component of “lithium” and “chloride” and takes the sum for the representation of the target.

Here, we propose a new means for the multiword phrase representation of chemical terms. Our Multiword Identifying and Representing (MIR) method starts with recognizing the multiword phrases in the chemical literature by an unsupervised data-driven model and then represent the identified phrases as new words in the vocabulary at the phrase level. Using the same example of representing “lithium chloride,” the MIR method treats this phrase as a new, independent word different from “lithium” and “chloride,” then represents these three terms separately. Through a series of designed experiments, the results indicated that the current phrase identification model sufficiently achieves the goal of obtaining a better representation of words. In addition, we showed that the MIR method has less information loss compared to the conventional approach in representing the multiword phrases. Our results demonstrated that the MIR method has the capability to accurately represent chemical terms and pave the road to improve the effectiveness and performance of downstream machine learning algorithm-incorporated tasks, where the representations of chemical terms will be used as learning features.

2. METHODOLOGY

2.1. Overview. The overall workflow of our approach and conventional approach is presented in Figure 1. Besides demonstrating the MIR method, we constructed a pipeline as the baseline model by employing the conventional method of representing multiword chemical terms. Both the conventional method and MIR method started with the same data source and tokenization. In the conventional method, word embedding is performed right after tokenization, and the representation of a phrase is obtained through a post-vector addition. In the MIR method, a new step is incorporated to identify multiword phrases and add the detected terms to the vocabulary. The word embedding is performed afterward at the phrase level. Finally, a series of experiments are designed to evaluate the representing performances of the conventional method and MIR method.

2.2. Data Source. The training data is composed of patents downloaded from the United States Patent and Trademark Office (USPTO). To focus on the chemistry domain, we kept patents that contain keywords of “lithium” and “synthesis”. A total of 119 166 patents were preserved as the training corpora to the current study. For comparison, the patents extracted with a cooperative patent classification code under the “chemistry” section had a total 356 612 patents. Thus, the training corpora used in this work served as good representatives of the chemistry domain.

2.3. Tokenization. The raw patent files were processed using the following procedures. First, each patent was segmented into sentences using the function sent_tokenize() from the NLTK library. After that, words were tokenized to keep necessary punctuation characters. The word tokens were then converted into lower case, and stop words were removed according to the stop words list from NLTK. We purposely kept those elemental names coincide with the stop words such as “I” as iodine, “At” as astatine, and “O” as oxygen. Finally, numeric values were removed from the tokens. We also removed ambiguous tokens such as “a.11.b.2.i” using several regular expression filters.

The tokenization should preserve tokens that contain valid punctuation characters such as the chemical identifiers, IUPAC name, InChi key, and SMILES. Instead of simply chopping sentence on whitespace and removing all punctuation characters from word tokens, we used several punctuations as token delimiters, including the whitespace symbol, the forward slash symbol, the equal symbol, and the number symbol. After that, a subset of punctuation characters frequently appeared in chemical terms were put into a whitelist to allow valid tokens to pass through. In some special cases when punctuation characters were confusing with sentence...
boundaries such as “degree.” and “no.”, we trimmed them out of the tokens.

### 2.4. Multiword Phrase Detection

There are multiple systems to identify multilword chemical phrases from chemistry literature, which can be categorized as dictionary-based, rule-based, supervised machine learning-based, unsupervised machine learning-based, and hybrid systems. Dictionary-based and rule-based systems require manually preparation based on public resources. Supervised machine learning-based systems require feature-based representation of sample data under human preparation and supervision. We chose an unsupervised statistical model for its capability to leverage statistical information and linguistic features and do the training without human annotation or supervision. In the unsupervised method, many measurements have been previously created to identify phrases in the text, such as mutual information (MI), point mutual information (PMI), and the normalized version of them. These methods, however, have a common drawback of heavy computing requirement. We adopted a data-driven phrase identification function proposed by Mikolov et al. Compared to other models, this data-driven method is able to identify not only chemical name phrases (e.g., compound names) but also some other kinds of phrases such as operation names, brand names, equipment names, etc. Meanwhile, it has an acceptable phrase identification rate, which brings additional advantages in terms of computational efficiency. The multiword identification function started from tokenized and trimmed single words (unigrams) in the sentence context obtained from the previous tokenization step. A scoring function from a third-party unsupervised semantic modeling python library Gensim was implemented to identify reasonable phrases (bigrams) formed by two co-occur unigrams. The bigram scoring function is defined as

\[
S = \frac{(\text{bigram\_count} - \text{min\_count})^{\text{len\_vocab}}}{(\text{unigram\_a\_count}\times\text{unigram\_b\_count})}
\]

where unigram\_a\_count and unigram\_b\_count are numbers of occurrences of the first and second component words. bigram\_count is the number of co-occurrences of the first and the second component words that are composing a phrase. len\_vocabulary is the vocabulary minimum collocation count threshold of a bigram among all patents. If S passes the predefined threshold value of s, the bigram is determined as a valid vocabulary phrase and added to the vocabulary. Phrases consisting of more than two words such as three-word phrases (trigrams) and four-word phrases (quadrigrams) are aggregated from one unigram one bigram and two bigrams, respectively. In principle, repeating this process could identify multiple-word phrases composed of n single words (n-grams, n is up to infinity), although it would be a memory-intensive task. As suggested by Mikolov et al., phrases composed of two to four words are a typical selected range for the training section. We, therefore, limited the detection to phrases containing no more than four words.

### 2.5. Word Embedding

After establishing the training corpus either directly from tokenization as the conventional approach does or after adding identified phrases to the vocabulary as the MIR approach does, we trained the word embedding model to map every word \( w \) from the vocabulary \( V \) to a numerical vector representation \( \vec{w} \) as \( w \rightarrow \vec{w} \in \mathbb{R}^D \) for all \( w \) in \( V \), where \( D \) is the dimensionality of representation \( \vec{w} \). We used the Word2Vec model for the word embedding. The original paper of Word2Vec model reported two architectures of training, the continuous bag-of-word (CBOW) and skip-gram (SG). Compared to other word embedding techniques, the CBOW architecture is more robust and better performed in relatedness and analogy tasks. Continuous bag-of-words (CBOW) structures were employed for all of the representation works in entire study.

The CBOW model is trained by maximizing the probability of the center word being \( w_c \) (“target” word to predict) if all of the surrounding words \( w_{-m} \rightarrow w_{-1} \rightarrow w_{c+1} \rightarrow w_{c+m} \) (training words) are given in terms of a softmax function, where \( m \) is the distance between center word \( w_c \) and the farthest context word

\[
P(w_{-m}w_{-m-1}...w_{c-1}w_{c+1}...w_{c+m} | w_c) = \text{softmax}(\text{score}(w_{-m}, ..., w_{c-1}, | w_{c+1}, ..., w_{c+m})))
\]

\[
= \log \frac{\exp (\text{score}(w_{-m}, ..., w_{c-1}, | w_{c+1}, ..., w_{c+m})))}{\sum_{m'} \exp (\text{score}(w_{-m'}, ..., w_{c-1}, | w_{c+1}, ..., w_{c+m'})))}
\]

where score \((w_{-m}, ..., w_{c-1}, w_{c+1}, ..., w_{c+m})\) is the compatibility of word \( w_c \) with surrounding words. A dot product is often used here as the score function. The objective function of this model is to minimize its negative log-likelihood on the training dataset

\[
\text{minimize}f = -\log P(w_{-m}w_{-m-1}...w_{c-1}w_{c+1}...w_{c+m})
\]

Because of the large size of training vocabulary \( V \), the sum operation in the denominator of the softmax function becomes very expensive. Two replacing techniques are introduced to improve the efficiency: hierarchical softmax and negative sampling. In addition, we used the subsampling technique to balance between frequently appearing and rarely appearing words. The implementation of word embedding used the Python library Gensim.

### 2.6. Post-Vector Addition

In the MIR approach, the word embedding directly outputs the representations for multiword phrases \( W_p \) for downstream tasks. However, in the conventional approach, the word embedding only outputs the representation for a single word. To represent a multiword phrase, a common way is to use the arithmetic aggregation of individual representations

\[
W_p = \sum_{i=0}^{n} W_i
\]

where \( W_p \) is a vector summed from \( n \) component word vectors and \( W_i \) is the representation for each component word.

### 2.7. Evaluation

To assess the performance of the representation of \( W_p \) from the conventional approach and the representation of \( W_p \) from the MIR approach, we created a total of six datasets for a series of evaluations. The first four datasets contain names of chemical compounds extracted from PubChem, a database of chemical molecules and their activities against biological assays maintained by the National Center for Biotechnology Information. Since we filtered the patent sources by only keeping those containing keywords of “lithium” and “synthesis”, the first dataset (D1) was designed to contain the names of 50 lithium organic and inorganic compounds. The second (D2), third (D3), and fourth (D4) dataset contained the names of inorganic compounds formed by two word, three word, and four word, respectively, with those overlapping with D1 removed. To evaluate the
performance of representing lithium-related and non-lithium-related chemical phrase terms, the fifth (D5) and sixth (D6) dataset were manually crafted by pulling phrases from the Wikipedia pages about “lithium battery” and “unit operations & separation process”. These crafted datasets were continuously used throughout our study, and the full lists of these datasets are provided in the Supporting Information.

3. RESULTS

3.1. Phrase Detection. First, we evaluated the capability of the MIR approach to correctly identify the chemical phrases. As shown in the Methodology section, a joint word is positively identified as a phrase if the score from eq 1 exceeds a manually defined threshold t. In practice, it is recommended to use a smaller threshold for phrases with longer sizes. However, to the best of our knowledge, there is no established rule on setting the threshold value to maximize the identification rate while keeping reasonable computational cost. We, therefore, used different t values in the current study. For each t value, we calculated the rate of successfully detecting the phrases in the evaluation datasets, as reported in Figure 2. For the min_count parameter, we used 30 in our study to keep a sufficient amount of bigram phrase candidates while maintaining the computational cost at an affordable level.

From the results, we verified that the MIR approach detects most of the bigrams in the evaluation datasets. All of the models with \( t \leq 1.25 \) achieve an identification rate higher than 50% on D1 and D2, and the highest identification rate 86% is scored on D5. For trigrams (D3) and quadrigrams (D4), all of the models achieve lower detection rates no matter how their threshold values were set, because the trigram and quadrigram chemical compound phrases appear less frequently than unigram or bigram compound names, hence making the denominator of the scoring function much larger than the enumerator. Nonetheless, the detection rate for these two datasets is still higher than 30%, indicating a decent success to find these high order phrases.

An important observation from Figure 2 is that the detection rate increased with the number of phrases in the vocabulary. This trend is not surprising, since the more phrases we detected, the more chance we find the correct one. In an extreme scenario, assuming that we retain every mult-word phrase as long as it appears in the source text, we should achieve the perfect detection rate while also obtaining the largest possible vocabulary. A large vocabulary means more computational demanding for the word embedding as well as other downstream tasks. Thus, it is important to balance the tradeoff between phrase detection rate and the total amount of phrases by appropriately tuning the threshold of \( t \). From Figure 2, we see that after a rapid increase for larger \( t \) values, the detection rate was nearly plateaued starting from \( t = 0.75 \) to 0.25. We used \( t = 0.75 \) for the rest of our studies. Additionally, the main results of our study were also verified using \( t = 0.25 \).

3.2. Chemical Name and Formula Relatedness Scoring. At this point, a new means to represent phrases from the chemistry-domain had been developed. We trained word representations through this new means. Better representations of phrases further contribute to the whole semantic vector space, which not only contains multiword phrases but also single-word vocabulary. We hypothesized that the connections between multword phrases and single words are expected to become closer and more obvious in the MIR approach. Therefore, a lot of applications can benefit from this new characteristic of the vector space. To verify our hypothesis, a series of experiments was designed to evaluate the representations generated through conventional approach and MIR approach.

The existing schemes to evaluate vector representations trained on the general domain of textual resources such as Wikipedia and Google News can be split into intrinsic and extrinsic evaluation in general. In intrinsic evaluation, word embeddings are evaluated by measuring performance among themselves in relatedness, analogy, categorization, and selectional preference tasks. In extrinsic evaluation, vector representations are used as input features to downstream tasks and their performance is evaluated based on these tasks. In the current study, we performed the intrinsic evaluation of the possible enhancements of the vector representations brought by the MIR method.

First, we consider the task of measuring the similarity between the name of a compound and its corresponding formula. This task is designed to examine whether the representations embed the precise semantic relationships or distances among chemical terms. For example, a two-word phrase of “lithium chloride” contains the same chemical meaning as a single word of “LiCl”. A good representation should, thus, represent the vector for “lithium chloride” similar to that of “LiCl”. This relatedness can be quantitively assessed using the cosine similarity function

\[
\text{similarity}(w_i, w_j) = \frac{\vec{w}_i \cdot \vec{w}_j}{\|w_i\| \|w_j\|}
\]

Datasets used in this experiment were obtained from D1, D2, and D3. We chose the ones with both chemical name and corresponding formula included in the vocabularies of two approaches. As shown in Figure 3, the relatedness score for the \( W \) representation varies between 0.4 and 0.5. By using the MIR approach, it is increased to around 0.6, suggesting that the \( W_p \) representation captures the semantic relationship between the chemical name and formula more accurately than \( W_c \). The highest similarity score was obtained for D1 at 0.65.

3.3. Chemical Name and Formula Inferring. While the previous experiment directly examined the similarities between
a given pair of chemical name and formulas, we designed another experiment to examine if the corresponding formulas can be correctly inferred by searching the geometric neighbors of chemical names. For example, providing the name of “lithium chloride”, the task is to look up the neighbor words ranging from 0 to vocabulary size $V$ to yield the correct answer “LiCl”. In principle, a good representation should rank semantically similar words to a target word higher, while rank less similar words lower, thus enabling the discovery of the correct formula in the vicinity of the corresponding chemical name.

We considered two ways to define the range of neighborhood. In the first run, we looked for the correct chemical formula within the first 20, 50, and 100 neighbors of a chemical name in two vector spaces. It, therefore, evaluated whether the formula is correctly located in a given number of neighbors of the target chemical name without considering the influence of local word density. For a dense population around a given word, the target word may remain in the vicinity but fall well outside of the detection range. Thus, in the second test, we looked for the formula within a given radius of similarity to the given chemical name. Results for this experiment are presented in Figure 4. We see that the $W_p$ representation performed poorly on this inferring task, especially for the compound names in D1. On the contrary, for the $W_p$ representation, the inferring task achieved a much better success rate with large portions of examples from D1 and D2 being correctly inferred. These two experiments, thus, concluded that semantic relationships between two similar/identical terms are better captured and presented by $W_p$ than $W_c$.

3.4. Synonyms, Acronyms, and Abbreviations Finding. Other ways to verify our hypothesis are the synonym finding and acronym/abbreviation finding tasks. In previous sections, we demonstrated how MIR contributes to finding the formula of a given chemical name. Here, we show that the same procedure can be applied here to find synonyms and acronyms/abbreviations based on the representations generated through MIR.

For synonym finding, we used $W_p$ to generate the nearest neighbors for 30 keywords (vocabulary-included) randomly selected from D1, D5, and D6 and checked if the neighbors included the correct synonyms (see full results in the Supporting Information). Table 1 shows several examples from this experiment.

It is apparent that synonyms were found for all of the words evaluated. Furthermore, not only the synonyms were found but also some related concepts or representatives were found such as “power tools,” “laptop computer,” and “cell phone” were found for word “portable electronics.” A core component of “lithium plating” was found, which is “sei layer.” These findings indicate that it is feasible to find synonyms through our MIR method only based on the similarities.

For the acronym/abbreviation finding, we first extracted 40 vocabulary-included acronyms/abbreviations of chemical terms from a reference book of chemistry visualization. Out of these testing terms, we found that within the range of 250 nearest neighbors, the correct acronyms/abbreviations were detected for 21 terms, as listed in Table 2. This decent success rate indicates that correct acronyms/abbreviations for a given term can be identified solely utilizing the similarity between the representations. It is, therefore, anticipated that better success rate can be achieved by combining the MIR representation with other established methods such as monolingual syntax-based methods or statistical information-based methods for synonym finding and string matching or natural language processing for acronyms/abbreviations finding.

3.5. Chemical Terms Clustering. We further evaluated if the representation of $W_p$ preserved the semantic relation between different phrases. For this purpose, we designed the experiment to examine the capability of recovering a clustering of labeled keywords into separate groups. After the phrase

![Figure 3. Relatedness scores on compound names and formulas as measured by cosine similarity.](image)

![Figure 4. Inferring results on compound names and formulas. (a) Searching formulas within a given number of nearest neighbors. (b) Searching formulas within a given range of cosine similarities.](image)
identification procedure, the model with threshold value $t$ set to 0.75 identified 25, 617, 41, 4, 30, 26 phrases out of $D_1$, $D_2$, $D_3$, $D_4$, $D_5$, and $D_6$, respectively. To balance between separate groups, we sampled 24 identified phrases each from $D_1$, $D_5$, and $D_6$. For the easiness of implementation and the capability of handling different kinds of similarities or distances, Hierarchical agglomerative clustering (HAC) was selected and performed to create a hierarchy of these keywords clusters according to the Euclidean distance metric (ward linkage), which was implemented through python Scikit-learn library. Euclidean similarity measures the magnitude (frequency) over the direction (semantic) of word vectors, and it can maximize the intercluster distance comparing to cosine similarity.\(^4\)\(^6\)\(^7\) Agglomerative (bottom-up) algorithm treats each document as a singleton cluster at the beginning, then successively combines nearby pair of document clusters into one cluster. Eventually, all clusters will be merged into one big cluster, which contains all of the documents. To visualize the clustering results, we plotted part of the dataset phrases in the principle component space, as shown in Figure 5. We find that the $W_p$ representation impressively recovered the correct clustering. Only the phrase of “lithium salt” and “charging speed” were clustered to the groups of compound names and operation, respectively. On the other side, the $W_c$ representation had much more phrases that were mistakenly grouped and a closer examination suggested that in the $W_c$ representation, the phrases from the compound name group and “lithium battery”-related group are difficult to distinguish from each other.

Quantitatively, the clustering results were evaluated by external evaluation metrics, including the mutual information-based score (MI),\(^6\)\(^0\) adjusted rand index (ARI),\(^5\)\(^9\) homogeneity, completeness, V-measure\(^6\)\(^0\) and Fowlkes-mallows index (FMI).\(^6\)\(^1\) As the scoring metrics in Table 3 indicate, the clustering performance based on representation $W_p$ greatly surpasses the clustering performance based on representation $W_c$. The external clustering measurement scores in other domains indicated that these metrics are normally ranging from 0 to 0.8, and the best clustered result can approach to 0.9.\(^6\)\(^2\)–\(^6\)\(^4\) Although it is not fair to compare our result with results from these works, which were measured based on completely different and integral datasets, our clustering result can still suggest that $W_p$ represent the semantic meanings among groups of phrases more accurately than conventional ones.

### 4. DISCUSSION

The evaluation results prove the capability of the MIR approach on effectively and accurately identifying and representing several types of multiword chemical phrases. Comparing to the conventional approach, less information loss is achieved in the MIR approach.
We analyzed why W_p performed better than W_c representation on the evaluation tasks. In W_c, the representation of a phrase is constructed through individual components, each of which is trained on the single-word level from its own context. For example, to obtain the representation of "lithium battery", we need to represent "lithium" by learning its co-occurrence with "chloride" when mentioned as a chemical name in the text resources, with "silvery-white" when describing the physical occurrence and with "battery" when describing in a specific application. Apparently, the information from the first two context serves as unnecessary noise in deciphering the actual meaning of "lithium" in "lithium battery". Same situation happens when representing "battery".

On the contrary, in the W_p representation, the phrase of "lithium battery" is treated as an independent word that differs with "lithium", "lithium chloride", and others, thus minimizes the influences from noisy background contexts.

Another information loss in the W_c occurs in summing the representations of individual words with equal weight. For example, we should anticipate that the phrase of "lithium battery", which describes a specific type of battery, to be more similar to "battery" than to "lithium". In other words, the representation of "battery" should have a higher weight than "lithium" in "lithium battery". On the contrary, equally weighting "lithium" and "battery" for the representation of "lithium battery" suggests that "lithium battery" is almost

---

**Table 3. External Clustering Evaluation Results of Hierarchical Agglomerative Clustering**

|       | MI    | ARI   | homogeneity | completeness | V-measure | FMI   |
|-------|-------|-------|-------------|-------------|-----------|-------|
| W_c   | 0.5948| 0.5675| 0.6055      | 0.6702      | 0.6362    | 0.7230|
| W_p   | 0.8910| 0.9170| 0.8938      | 0.8953      | 0.8946    | 0.9439|

---

**Figure 5.** Visualization of clustering results on part of the samples. Only first two principal components of the representations were used to build the graph. Color graded represents clusters predicted by the hierarchical agglomerative algorithm. Black arrow and empty symbols represent wrongly clustered samples compared to the ground truth. (a) Partial of clustering results based on W_c representation. (b) Partial of clustering based on W_p representation. (c) Names of the examined phrases. The samples from D1, D5, and D6 datasets are represented as uppercase letters, lowercase letters, and numbers, respectively.

**Figure 6.** Illustration of relationships between W_c, W_p, and their constituent word vectors in two vector spaces. Cosine similarities are measured from real models. (a) In W_c representation, "lithium battery" is almost equally similar to "battery" and "lithium". (b) In W_p representation, "lithium battery" is more similar to "battery" than to "lithium".
equally similar to “battery” and “lithium” (Figure 6a). By treating “lithium battery” as an independent phrase and training the embedding at the phrase level, the preservation of semantic meaning is improved. As a result, the representation of “lithium battery” is closer to “battery” than to “lithium” in the \( W_p \) representation (Figure 6b).

As we have shown in Section 3.1, a critical factor in our MIR method is the occurrences from eq 1 that determines the identification of a phrase in the text resource. One restriction on the occurrences is the threshold of the minimum appearance of a phrase (min_count). For a phrase with a number of appearances less than the min_count, eq 1 yields a negative score and hence directly rejects to add it to the vocabulary. However, there are certain circumstances that chemical terms only appear a few times in the text resources but should be considered as a phrase, especially for some compound names such as “lithium magnesium silicate”. This limitation of phrase detection function can be alleviated by introducing chemical databases and third-party tools for looking up. We anticipate the dictionary-based, rule-based, or hybrid methods in cooperation with the current proposed MIR approach could result in a more accurate detection of chemical phrases without significantly sacrificing computing resources.

There are still several limitations of our work. One of them is the noise phrase generated accompanying identifying real chemical multiword phrases. Although improvements to word representations achieved by moving the phrases identification procedure forward had already been verified under the current model settings, it is reasonable to infer that the performance will become better if more noise phrases were removed from the current context. As mentioned before, some additional approaches, such as dictionary-lookup and rule-based parsing, can be incorporated with MIR in the future to help create a “clean” context to get better word embeddings. Another limitation of our current study is the lack of extrinsic evaluations. One purpose of word representation is to create a numerical vector space for the downstream machine learning tasks. From this point of view, the evaluation of the performance of representation should also be thoroughly investigated in these downstream jobs. While the current study has extensively evaluated the semantic relatedness in the MIR approach, it is of great interest to see how the MIR representation performs as input in future downstream machine learning models.

5. CONCLUSIONS

To summary, the current work introduces the multiword identifying and representing method, a fine-tuned workflow to detect multiword chemical terms in the chemical literature. The MIR starts with identifying multiword phrases, adding it to the vocabulary, and performing the word embedding at the phrase level. Through a series of special-designed experiments, we demonstrated that the MIR approach effectively and accurately identifies multiword chemical terms from chemical patents. Compared to the conventional approach representing constituent single words first and combining them afterward, the MIR method is more robust and accurate to preserve the semantic meaning of multiword chemical phrases. The accurate representation of chemical terms is the first and essential step to provide learning features for downstream natural language processing tasks. The MIR method, thus, paves the road to utilize a large volume of chemical literature in future data-driven studies.

ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acsomega.9b02060.

AUTHOR INFORMATION

Corresponding Author

*E-mail: chen.ling@toyota.com.

ORCID

Chen Ling: 0000-0002-8667-9792

Notes

The authors declare no competing financial interest.

REFERENCES

(1) Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; Chen, Y.; Lillicrap, T.; Hui, F.; Sifre, L.; van den Driessche, G.; Graepel, T.; Hassabis, D. Mastering the game of Go without human knowledge. Nature 2017, 550, 354.

(2) Esteva, A.; Kuprel, B.; Novoa, R. A.; Ko, J.; Swetter, S. M.; Blau, H. M.; Thrun, S. Dermatologist-level classification of skin cancer with deep neural networks. Nature 2017, 542, 115.

(3) Krizhevsky, A.; Sutskever, I.; Hinton, G. E. Imagenet Classification with Deep Convolutional Neural Networks, Advances in Neural Information Processing Systems, 2012; pp 1097–1105.

(4) Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning To Align and Translate. arXiv preprint arXiv:1409.0473, 2014.

(5) Butler, K. T.; Davies, D. W.; Cartwright, H.; Isayev, O.; Walsh, A. Machine learning for molecular and materials science. Nature 2018, 559, 547.

(6) Mueller, T.; Kunse, A. G.; Ramprasad, R. Machine learning in materials science: Recent progress and emerging applications. Rev. Comput. Chem. 2016, 29, 186–273.

(7) Ward, L.; Agrawal, A.; Choudhary, A.; Wolverton, C. A general-purpose machine learning framework for predicting properties of inorganic materials. npj Comput. Mater. 2016, 2, 16028.

(8) Faber, F.; Lindmaa, Å.; von Lilienfeld, O. A.; Armiento, R. Crystal structure representations for machine learning models of formation energies. Int. J. Quantum Chem. 2015, 115, 1094–1101.

(9) Chen, C.; Ye, W.; Zuo, Y.; Zheng, C.; Ong, S. P., Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals. arXiv preprint arXiv:1812.05055, 2018.

(10) Xie, T.; Grossman, J. C. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. Phys. Rev. Lett. 2018, 120, No. 145301.

(11) Jha, D.; Ward, L.; Paul, A.; Liao, W.-k.; Choudhary, A.; Wolverton, C.; Agrawal, A. ElemeNet: Deep learning the chemistry of materials from only elemental composition. Sci. Rep. 2018, 8, No. 17593.
(12) Li, Z.; Wang, S.; Chin, W. S.; Achenie, L. E.; Xin, H. High-throughput screening of bimetallic catalysts enabled by machine learning. J. Mater. Chem. A 2017, 5, 24131–24138.

(13) Li, Z.; Wang, S.; Xin, H. Toward artificial intelligence in catalysis. Nat. Catal. 2018, 1, 641–642.

(14) Ma, X.; Li, Z.; Achenie, L. E.; Xin, H. Machine-learning-augmented chemisorption model for CO2 electroreduction catalyst screening. J. Phys. Chem. Lett. 2015, 6, 3528–3533.

(15) Elton, D. C.; Boukouvalas, Z.; Fuge, M. D.; Chung, P. W. Deep Learning for Molecular Generation and Optimization: A Review of the State of the Art. arXiv preprint arXiv:1903.04388, 2019.

(16) Ghamene, G. L.; Sanchez-Lengeling, B.; Outeiral, C.; Farias, P. L. C.; Aspuru-Guzik, A. Objective-Driven Generative Adversarial Networks (ORGAN) for Sequence Generation Models. arXiv preprint arXiv:1705.10843, 2017.

(17) Benjamin, S.; Carlos, O.; Gabriel, L.; Alan, A. Optimizing Distributions over Molecular Space. An objective-driven generative adversarial network for inverse-design chemistry (ORGANIC). ChemRxiv, 2017.

(18) Fooohsee, D.; Mood, A.; Gutman, E.; Tavakoli, M.; Urban, G.; Liu, F.; Huyhn, N.; Van Vranken, D.; Baldi, P. Deep learning for chemical reaction prediction. Mol. Syst. Des. Eng. 2018, 3, 442–452.

(19) Schwalller, P.; Gaudin, T.; Lanyi, D.; Bekas, C.; Laino, T. "Found in Translation": predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. Chem. Sci. 2018, 9, 6091–6098.

(20) Agrawal, A.; Choudhary, A. Perspective: Materials informatics and big data: Realization of the "fourth paradigm" of science in materials science. APL Mater. 2016, 4, No. 053208.

(21) Jain, A.; Og, S. P.; Hautier, G.; Chen, W.; Richards, W. D.; Dacek, S.; Cholia, S.; Gunter, D.; Skinner, D.; Ceder, G.; et al. Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. APL Mater. 2013, 1, No. 011002.

(22) Saal, J. E.; Kirklin, S.; Akyol, M.; Meredith, B.; Wvoronov, C. Materials design and discovery with high-throughput density functional theory: the open quantum materials database (OQMD). JOM 2013, 65, 1501–1509.

(23) Curtarolo, S.; Setyanaw, W.; Hart, G. L.; Jahnatek, M.; Chepulsiki, R. V.; Taylor, R. H.; Wang, S.; Xue, J.; Yang, K.; Levy, O.; et al. AFLLOW: an automatic framework for high-throughput materials discovery. Comput. Mater. Sci. 2012, 58, 218–226.

(24) Zhang, L.; Bergstra, J. A strategy to apply machine learning to small datasets in materials science. npj Comput. Mater. 2018, 4, 25.

(25) Kim, E.; Huang, K.; Tomala, A.; Matthews, S.; Strubell, E.; Saunders, A.; McCallum, A.; Olivetti, E. Machine-learned and codified synthesis parameters of oxide materials. Sci. Data 2017, 4, No. 170127.

(26) Kim, E.; Huang, K.; Saunders, A.; McCallum, A.; Ceder, G.; Olivetti, E. Materials synthesis insights from scientific literature via deep learning. J. Phys. Chem. Lett. 2019, 10, 9436–9444.

(27) Jensen, Z.; Kim, E.; Kwon, S.; Gani, T. Z.; Rama’ė-Leshkov, Y.; Mithun, M.; Corma, A.; Olivetti, E. A Machine Learning Approach to Zeolite Synthesis Enabled by Automatic Literature Data Extraction. ACS Cent. Sci. 2019, DOI: 10.1021/acscentsci.9b00193.

(28) Kim, E.; Huang, K.; Jegelka, S.; Olivetti, E. Virtual screening of inorganic materials synthesis parameters with deep learning. npj Comput. Mater. 2017, 3, 53.

(29) Kim, E.; Huang, K.; Kononova, O.; Ceder, G.; Olivetti, E. Distilling a Materials Synthesis Ontology. Matter 2019, 8, 12–12.

(30) Kim, E.; Jensen, Z.; van Grootel, A.; Huang, K.; Stahl, M.; Myore, S.; Chang, H.-S.; Strubell, E.; McCallum, A.; Jegelka, S. Inorganic Materials Synthesis Planning with Literature-Related Neural Networks. arXiv preprint arXiv:1903.00302, 2019.

(31) Elton, D. C.; Turakhia, D.; Reddy, N.; Boukouvalas, Z.; Fuge, M. D.; Doherty, R. M.; Chung, P. W. Using Natural Language Processing Techniques to Extract Information on the Properties and Functionalities of Energetic Materials from Large Text Corpora. arXiv preprint arXiv:1903.00415, 2019.
(54) Van der Plas, L.; Tiedemann, J. Finding Synonyms Using Automatic Word Alignment and Measures of Distributional Similarity, Proceedings of the COLING/ACL on Main Conference Poster Sessions, 2006; pp 866–873.

(55) Schwartz, A. S.; Hearst, M. A. A Simple Algorithm for Identifying Abbreviation Definitions in Biomedical text. In Biocomputing 2003; World Scientific, 2002; pp 451–462.

(56) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V. Scikit-learn: Machine learning in Python. J. Mach. Learn. Res. 2011, 12, 2825–2830.

(57) Xu, R.; Wunsch, D. C. Survey of Clustering Algorithms, 2005.

(58) Strehl, A.; Ghosh, J. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. J. Mach. Learn. Res. 2002, 3, 583–617.

(59) Hubert, L.; Arabie, P. Comparing partitions. J. Classif. 1985, 2, 193–218.

(60) Rosenberg, A.; Hirschberg, J. V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure, Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), 2007.

(61) Fowlkes, E. B.; Mallows, C. L. A method for comparing two hierarchical clusterings. J. Am. Stat. Assoc. 1983, 78, 553–569.

(62) Priness, I.; Maimon, O.; Ben-Gal, I. Evaluation of gene-expression clustering via mutual information distance measure. BMC Bioinf. 2007, 8, 111.

(63) Yeung, K. Y.; Fraley, C.; Murua, A.; Raftery, A. E.; Ruzzo, W. L. Model-based clustering and data transformations for gene expression data. Bioinformatics 2001, 17, 977–987.

(64) Reichart, R.; Rappoport, A. The NVI Clustering Evaluation Measure, Proceedings of the Thirteenth Conference on Computational Natural Language Learning, 2009; pp 165–173.