

ES2Vec: Earth Science Metadata Keyword Assignment using Domain-Specific Word Embeddings

1st Muthukumar Ramasubramanian
Dept. of Computer Science
The University of Alabama in Huntsville
 Huntsville, USA
 mr0051@uah.edu

2nd Hassan Muhammad
Dept. of Computer Science
The University of Alabama in Huntsville
 Huntsville, USA
 hmm0008@uah.edu

3rd Iksha Gurung
The University of Alabama in Huntsville
 Huntsville, USA
 ig0004@uah.edu

4th Manil Maskey
NASA Marshall Space Flight Center
 Huntsville, USA
 manil.maskey@nasa.gov

5th Rahul Ramachandran
NASA Marshall Space Flight Center
 Huntsville, USA
 rahul.ramachandran@nasa.gov

Abstract—Earth science metadata keyword assignment is a challenging problem. Dataset curators select appropriate keywords from the Global Change Master Directory (GCMD) set of keywords. The keywords are integral part of search and discovery of these datasets. Hence, selection of keywords are crucial in increasing the discoverability of datasets. Utilizing machine learning techniques, we provide users with automated keyword suggestions as an improved approach to complement manual selection. We trained a machine learning model that leverages the semantic embedding ability of Word2Vec models to process abstracts and suggest relevant keywords. A user interface tool we built to assist data curators in assignment of such keywords is also described.

Index Terms—Word2Vec, Natural Language Processing, Keyword Classification, Machine Learning, Neural Network, Classifier.

I. INTRODUCTION

NASA's growing collection of Earth science datasets are described by metadata records stored in a catalog called the Common Metadata Repository (CMR) [1]. The CMR leverages the Global Change Mastery Directory (GCMD) [2] science keyword taxonomy, which is a hierarchical set of controlled Earth science keywords. GCMD Keywords are used to help ensure Earth science data, services, and variables are described in a consistent and comprehensive manner [3]. These science keywords are manually assigned to datasets using data providers' and curators' knowledge of the dataset abstracts present in their respective metadata records. This process involves a team of people assigning these keywords to the metadata record with the best of their knowledge about the data. Assigning keywords manually is labor intensive and

is prone to human error and inconsistencies. Thus, the error and inconsistencies propagate into the search and discovery of these datasets. Because the science keywords are vital to data discovery, there is a need for a reliable way to assign keywords to dataset.

Our proposed solution to this problem is leveraging machine learning to accurately assign science keywords to datasets in an automated, objective, and consistent manner. We developed a keyword classifier that takes word embeddings of the abstracts as input. The results of the classifier are suggested keywords along with their probability scores. Word2Vec embedding was built from the science corpus that captures linguistic and domain-specific relationships between words in the corpus. Our implementation was trained on a corpus of 21,318 Earth science related journal articles. We also built a web-tool that takes an abstract as input and returns relevant GCMD science keywords and accuracy metrics. Users can provide feedback on the model by indicating whether the assigned keywords are correct or incorrect. Our main contributions from this work are as follows:

- A web-tool that Subject Matter Experts (SMEs) can use and tag their datasets with appropriate keywords in a more robust and automated way.
- A domain-specific word embedding model that performs better than universal word embedding models on classification tasks related to Earth science domain.

The rest of the paper is organised as follows: the next section (II) provides background on various methods used in this paper. Section III discusses motivations and related work for the paper. We outline our methodology in section IV and compare classification results from different word embeddings in section V. We then briefly talk about our web-tool in section VI. Finally, we conclude the paper in section VII and discuss

This work is funded by NASA-IMPACT

future work in section VIII.

II. BACKGROUND

A. Artificial Neural Networks

Artificial Neural Networks (ANN) are computing systems whose structure and function are modeled after the neurons in a biological brain. They consist of three main layers: input, hidden, and output. The layers are sequentially connected. Each layer has multiple neurons that takes in input from previous layer. It activates the inputs by aggregating and transforming them using *activation functions*. The difference (*Error*) between activations from the output layer and expected output is calculated. This *Error* is minimized iteratively by updating weights between layers in the network. This is done using *Backpropogation* [4].

B. Word2Vec and Word Embeddings

Word2Vec [5] is a two-layer neural network used to construct vector representations of words called *word embeddings* from text. Word embeddings are shown [6] to break down words into latent variables, each of them capturing hidden relationships present in the corpus, as shown in Fig. 1. The two word representation methods commonly used to train Word2Vec are as follows:

1) *Skip Gram*: Predicts source context words from target words. Skip Gram works well with small amounts of data and performs well on rare words.

2) *Common Bag of Words (CBOW)*: Predicts target words from source context words. CBOW is faster than Skip Gram and performs well for more frequent words.

	Hot	Cold	Dry	Wet	Toxic	Plume
Smoke	0.9	-0.4	0.6	-0.8	0.7	0.8
Dust	0.5	0.2	0.7	-0.5	0.2	0.8
Ocean	0.3	0.5	-0.9	0.9	0.3	-0.7
Snow	-0.7	0.8	0.3	0.6	-0.3	-0.4

Fig. 1. Word Embedding Illustration

III. RELATED WORK

Using word embeddings from Word2Vec as an input embedding layer for classification tasks has been explored in numerous previous works. In 2014, Kim et. al. [7] showed that a 1-dimensional Convolutional Neural Network (CNN) that uses word embeddings from a pre-trained Word2Vec as input can perform well on multiple benchmarks. This suggests that the word embeddings are *universal feature extractors* and can be used for various classification tasks. Zhang et. al. [8] successfully classified user sentiment using Word2Vec to embed user comments from e-commerce websites and a variation of Support Vector Machine (SVM) as a classifier. To do that, they applied Word2Vec for clustering English words

with similar latent embeddings. Furthermore, they employed two feature selection methods: lexicon-based, to find words similar to words representing sentiments (using Word2Vec) and parts-of-speech based, to contrast and model sentiments represented by adverbs, verbs and adjectives compared to using just adjectives. Finally, an SVM classifier was used to classify the selected features into user sentiments.

In our work, we employed a similar approach to the aforementioned methods, except in our case, we trained an Earth science specific Word2Vec, here after referenced as ES2Vec (discussed in section IV-C). We used ES2Vec to obtain word embeddings for abstracts. The embeddings are used as input to a neural network, which was trained to assign relevant keywords to abstracts. we also built a web service for users to enter abstracts of Earth science data and obtain relevant keywords using the aforementioned method.

IV. METHODOLOGY

The methodology section is organised as follows: In Subsection IV-A, We explain our training and inference workflow. The training data used for our work is detailed in subsection IV-B. Subsection IV-C discusses ES2Vec in further detail and subsection IV-D elaborates on the fully connected neural network used.

A. Workflow

The workflow of our proposed system is shown in Fig. 2. It is broadly divided into two phases:

1) *Training Phase*: Abstracts for datasets and their respective human-tagged keywords were obtained from NASA Distributed Active Archive Centers (DAACs). The abstracts were converted to word embeddings using ES2Vec (Sec. IV-C) and the keywords were converted into one-hot vectors. A classifier was trained to map the word embeddings to the (one-hot) keyword space.

2) *Inference Phase*: New abstracts were embedded using ES2Vec (Sec. IV-C) and passed as input to the classifier. The one-hot keyword outputs from the neural network were converted back to their respective text format and presented to the user.

B. Training Data

1) *Input*: The input data are text abstract fields for datasets from three of the NASA Distributed Active Archive Centers (DAACs). Abstracts were cleaned using the following methods:

- Stopword Removal
- Stemming
- Tokenization with Phrase Retention

The first two cleaning methods are widely used in Natural Language Processing (NLP). There are several commonly used phrases in the Earth science domain and breaking them down into component words can lead to potential loss of context. To make sure that the phrases have word embeddings of their own, we used tokenization with phrase retention. The

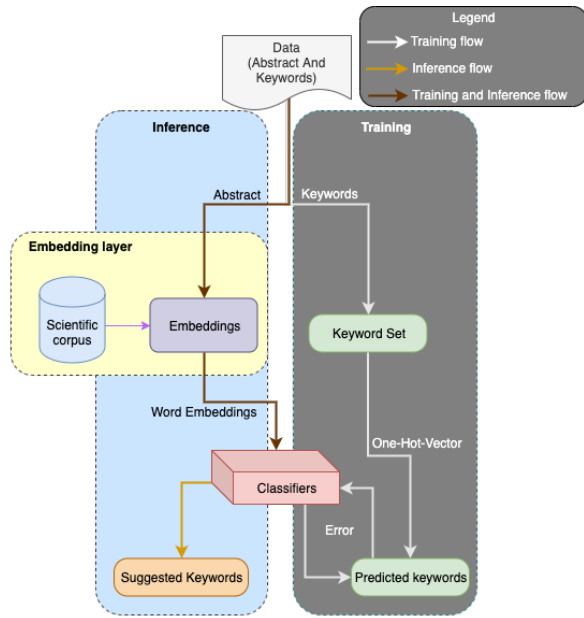


Fig. 2. GCMD Keyword Classification tool - Training/Inference Workflow

list of phrases prevalent in the Earth science fields were collected from Earth-science specific ontologies such as Semantic Web for Earth and Environmental Terminology SWEET [9], American Meteorological Society (AMS) Glossary [10], and GCMD Keywords [2] and added to the existing words in the vocabulary.

2) *Labels*: The output labels for abstracts are human-assigned Earth science keywords. The keywords for all the available abstracts (with frequency of occurrence ≥ 5) were collected and a unique set of keywords were formed. This unique keyword set (274 unique keywords) was used to index keywords and obtain their respective one-hot encodings, as illustrated in Fig. 3.

	Keywords							
Abstract 1	Forest	Biomass	Vegetation					
Abstract 2	Ozone	Radiation	Atmosphere					
Abstract 3	Aerosols	Atmosphere						

	One-hot Encoding							
Keyword	Aerosols	Ozone	Forest	Biomass	...	Radiation	Vegetation	Atmosphere
Abstract 1	0	0	1	1	...	0	1	0
Abstract 2	0	1	0	0	...	1	0	1
Abstract 3	1	0	0	0	...	0	0	1

Fig. 3. One-hot encoding of keywords

C. ES2Vec: Earth Science Specific Word Embeddings

Ghosh *et al.* [11] showed that a domain-specific word embedding model built using a disease based vocabulary outperformed other universal embedding models in capturing disease related taxonomy attributes. Sarma *et al.* [12] proposed a method that uses domain adapted embedding, a combination of universal embedding and domain specific embedding to perform sentiment classification tasks. We used these works as motivations to develop a highly domain-specific Word2Vec

called ES2Vec. It was trained using only Earth science vocabulary to perform a highly domain-specific keyword classification task. The source for the Earth science vocabulary is the 21,318 Earth science journals obtained from American Geophysical Union (AGU). The corpus was cleaned using methods discussed in Sec. IV-B1. We trained two Word2Vec models, with 150 and 300 as the embedding size respectively. Both the models were trained using the same aforementioned corpus. The corpus consisted of 115M words in which about 500k of them constitute the unique vocabulary set. We used *gensim* [13] Continuous Bag of Words (CBOW) model to train the Word2Vec, with window size of 10.

D. Neural Network Classifier

Our proposed neural network consists of 4 fully connected layers. The input to the network is the average of word embeddings of all the words present in an input abstract. Each of the nodes was activated by a Rectified Linear Unit (*ReLU*) [14] activation function. Each layer is followed by *Dropouts* [15], to improve the generalizability of the model by implicitly taking a weighted average of multiple similar models. The output of the model is the one-hot encoded keyword vector discussed in section IV-B2. The architecture of the model is illustrated in Fig. 4.

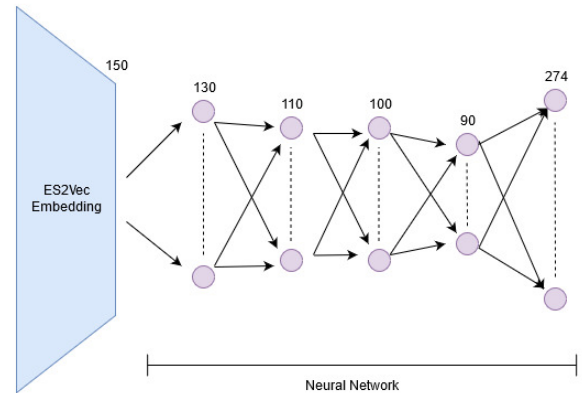


Fig. 4. Network Architecture

1) *Training*: This model uses an Adam (Adaptive Moment Estimation) [16] optimizer, a variation of stochastic gradient descent [17] by using moving average of the gradient in order to update the model's weights to achieve optimal values. We use binary cross-entropy loss function (L) (Shown in Equation 1), to measure the performance of the classification model.

$$L = \sum_{c=1}^{274} -y_c \log(p_c) + (1 - y_c) \log(1 - p_c) \quad (1)$$

Where,

L = Loss Function

c = class index

y_c = c^{th} target keyword

p_c = c^{th} predicted keyword

Adam finds the parameter values that minimizes the loss function L . The model was trained iteratively, and for every iteration, the weights are updated such that their contribution to loss function L is reduced. This model was trained on 2078 samples and validated on 520 samples. The model continuously trains until there was no decrease in validation loss for 10 consecutive epochs and the snapshot of the model configuration (model weights) at its lowest validation loss among all the epochs was chosen as the final model.

E. Accuracy Metric

Accuracy score is defined as the fraction of the number of keywords correctly predicted over the total number of assigned keywords, as shown in Eq. (2). The *threshold* represents the confidence of the model in assigning a keyword to given text. For our experiments, we empirically chose *threshold* = 0.15.

$$score = \sum_{i=1}^n [x_i \geq threshold] * X / \sum_{i=1}^n X \quad (2)$$

Where,

n = total number of keywords

x_i = predicted probability of keyword at i^{th} index

X = target one-hot keyword vector

V. RESULTS

To highlight the effectiveness of ES2Vec in performing domain-specific tasks, we trained multiple keyword classification models, each differing only by the Word2Vec model used for word embedding. All other parameters, including the number of nodes and layers of the fully connected neural network were kept unchanged. For comparison, the embedding weights were fixed and rest of the weights of the model were trained freely. The universal word embedding models used to compare our ES2Vec model are listed as follows:

- **glove-twitter-25, glove-twitter-50, glove-twitter-200, glove-wiki-gigaword-100**

GloVe (Global Vectors for Word Representation) [18] is an unsupervised algorithm which uses statistics based on word-word co-occurrences in text for obtaining word embeddings. The vector representations obtained from GloVe has tendency to capture interesting linear relationships between words. For this experiment, we used four different versions of GloVe. The first three were trained on twitter corpus with 25, 50, and 200 embedding dimensions respectively. The fourth model was trained on text from Wikipedia articles published for the year 2014 and fifth edition of english Gigaword [19].

- **Word2Vec-google-news-300**

These are pre-trained word vectors based on [6] and obtained from Google news articles containing about 150 Billion words. The model contains a total of 300M words embedded into 300 word vectors.

- **fasttext-wiki-news-subwords-300**

This Word2Vec was trained using text from wikipedia articles from the year 2017, combined with text from UMBC webbase corpus [20] and news data from *statmg.org*, a website dedicated for statistical translation of human languages [21]. The embedding dimension of this model is 300.

The performance of the models were evaluated using the accuracy metric described in section IV-E for 520 validation samples. The results are given in Table I.

Word2Vec model	Embedding size	Vocabulary size	No. of Tokens	Accuracy
glove-twitter-25	25	1.2M	27B	0.596
glove-twitter-50	50	1.2M	27B	0.657
glove-twitter-200	200	1.2M	27B	0.679
fasttext-wiki-news-subwords-300	300	1M	16B	0.713
Word2Vec-google-news-300	300	3M	100B	0.714
glove-wiki-gigaword-100	100	400k	6B	0.728
ES2Vec-150	150	500k	115M	0.786
ES2Vec-300	300	500k	115M	0.794

TABLE I
WORD2VEC MODEL PERFORMANCE FOR KEYWORD CLASSIFICATION TASK

From Table I, It is shown that our domain specific embedding model ES2Vec outperformed other universal embedding models in the highly domain-specific keyword classification task presented in this paper. Subjectively, it is also observed that the universal embeddings with higher chances of having science-related embeddings (e.g. glove-wiki-gigaword-100) performs better than the ones having lower chance of having science-related embeddings (e.g. glove-twitter-25). It is noteworthy that both the versions of ES2Vec performs better than other models, while using significantly fewer number of tokens and comparable vocabulary size. This indicates avenue for more performance improvements by adding more data to the corpus.

VI. WEB APPLICATION

One of the primary design goals of developing this model was to give the SMEs a more objective way of assigning science keywords to abstracts in metadata records. An elegant method to accomplish this task is through a simple interface where the SME inputs their abstract text, and immediately gets back relevant keywords for the abstract classified by the model. We built the GCMD Keyword Classification Tool to precisely fulfill this design requirement.

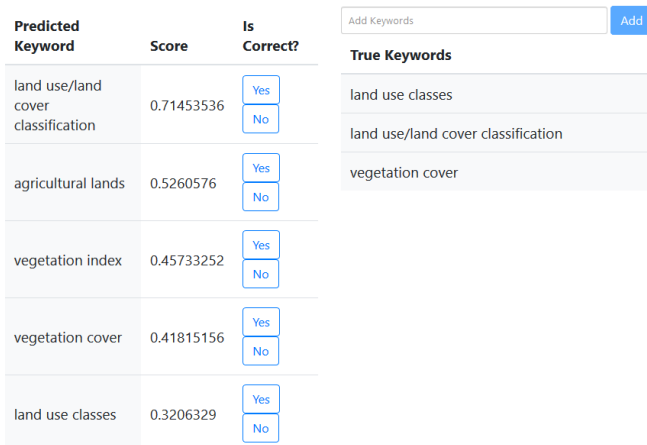
A. GCMD Keyword Classification Tool

The GCMD Keyword Classification Tool is a web application built on top of the ES2Vec classifier to suggest science keywords from an input abstract. This tool accepts two types of input:

- 1) Free-form text: Where the user would type or paste their abstract into the description field.

Description

The objective of the International Satellite Land Surface Climatology Project (ISLSCP II) study that produced this data set, ISLSCP II University of Maryland Global Land Cover Classifications 1992-1993, was to create a land cover map derived from 1 kilometer Advanced Very High Resolution Radiometer (AVHRR) data using all available bands. During this re-processing, the original University of Maryland (UMD) land cover type and fraction maps were adjusted to match the water/land fraction of the ISLSCP II land/water mask. These maps were generated for use by modelers of global biogeochemical cycles and others in need of an internally consistent, global depiction of land cover. This product describes the geographic distributions of 13 classes of vegetation cover (plus water and unclassified classes) based on a modified International Geosphere-Biosphere Programme (IGBP) legend.



Predicted Keyword	Score	Is Correct?
land use/land cover classification	0.71453536	<input type="button" value="Yes"/> <input type="button" value="No"/>
agricultural lands	0.5260576	<input type="button" value="Yes"/> <input type="button" value="No"/>
vegetation index	0.45733252	<input type="button" value="Yes"/> <input type="button" value="No"/>
vegetation cover	0.41815156	<input type="button" value="Yes"/> <input type="button" value="No"/>
land use classes	0.3206329	<input type="button" value="Yes"/> <input type="button" value="No"/>

Add Keywords

True Keywords

- land use classes
- land use/land cover classification
- vegetation cover

Fig. 5. GCMD Keyword Classification Tool results section.

- 2) Concept-id: Where the user would enter a concept-id (id uniquely identifying Earth science datasets maintained at NASA DAACs) into the concept-id field and the corresponding abstract related to the dataset pointed by the concept-id would be used by the classifier as input for keyword suggestions.

Each concept-id also has versioning, where the user would select the version they would like to get suggested keywords from a list of available versions of the dataset metadata record. If no version is selected, the latest available version is used. Once the user decides on their method of input, they would click the *Suggest Keywords* button.

The predicted keywords and relevant scores for the description is returned from the classifier back to the tool. The resulting webpage as shown in Fig. 5 consists of the submitted abstract, predicted keywords, and relevant scores. If the user had provided a concept-id, the keywords of the corresponding dataset in CMR are scraped and shown in conjunction with the predicted keywords. The user can validate the correctness of the predicted keywords by selecting the *Yes* or *No* buttons next to each keyword in the predicted keywords table. In the future, this feedback will serve as new training data that will improve the machine learning model. The GCMD Keyword Classification Tool can be accessed by visiting <https://gcmd.nasa-impact.net/>.

VII. CONCLUSION

In this paper, we have presented a domain-specific word embedding model that performs better than universal word embedding models on classification tasks related to Earth science domain. In addition, a web-tool that uses the aforementioned embedding to assign keywords to datasets is presented. The results show that domain constrained Word2Vec performs better than universal embedding for specific use cases. This resulted in SMEs assigning science keywords that were consistent and more relevant to the dataset. We found it to be difficult to construct domain specific corpus.

VIII. FUTURE WORK

Some of the tasks planned as future work are as follows: (1) enrich ES2Vec model by increasing both the size of vocabulary and the number of tokens with addition of more Earth science related articles (including news sources) to the corpus, (2) train the word embedding weights along with classifier weights and analyse performance of the classifier, (3) as shown in Sarma et. al [12], investigate ways to incorporate generic embeddings along with ES2Vec to potentially improve accuracy of the model.

REFERENCES

- [1] "Common Metadata Repository (CMR)," <https://earthdata.nasa.gov/eosdis/science-system-description/eosdis-components/cmr>, 2019.
- [2] "Global Change Master Directory (GCMD)." Greenbelt, MD: Global Change Data Center, Science and Exploration Directorate, Goddard Space Flight Center (GSFC) National Aeronautics and Space Administration (NASA), 2018. [Online]. Available: URL (GCMD Keyword Forum Page): <https://earthdata.nasa.gov/gcmd-forum>
- [3] "GCMD Keywords," <https://earthdata.nasa.gov/earth-observation-data/find-data/gcmd/gcmd-keywords>, webpage.
- [4] R. Rojas, *The Backpropagation Algorithm*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996, pp. 149-182.
- [5] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," 2013.
- [6] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," 2013.
- [7] Y. Kim, "Convolutional neural networks for sentence classification," *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014. [Online]. Available: <http://dx.doi.org/10.3115/v1/D14-1181>
- [8] D. Zhang, H. Xu, Z. Su, and Y. Xu, "Chinese comments sentiment classification based on word2vec and svmperf," *Expert Systems with Applications*, vol. 42, no. 4, pp. 1857-1863, 2015.
- [9] R. Raskin, "Sweet 2.1 ontologies," AGU Fall Meeting Abstracts, 2010.
- [10] "AMS glossary," <http://glossary.ametsoc.org/wiki/Special:AllPages>, 2019.
- [11] S. Ghosh, P. Chakraborty, E. Cohn, J. S. Brownstein, and N. Ramakrishnan, "Characterizing diseases from unstructured text," *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management - CIKM '16*, 2016. [Online]. Available: <http://dx.doi.org/10.1145/2983323.2983362>
- [12] P. K Sarma, Y. Liang, and B. Sethares, "Domain adapted word embeddings for improved sentiment classification," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 37-42. [Online]. Available: <https://www.aclweb.org/anthology/P18-2007>
- [13] R. Rehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Valletta, Malta: ELRA, May 2010, pp. 45-50, <http://is.muni.cz/publication/884893/en>.

- [14] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814.
- [15] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.
- [16] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014.
- [17] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*. Springer, 2010, pp. 177–186.
- [18] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543. [Online]. Available: <http://www.aclweb.org/anthology/D14-1162>
- [19] R. Parker, "English gigaword fifth edition ldc2011t07," <https://catalog.ldc.upenn.edu/LDC2011T07>, web download.
- [20] L. Han, A. L. Kashyap, T. Finin, J. Mayfield, and J. Weese, "Umber ebiq: Semantic textual similarity systems," in *Proceedings of the Second Joint Conference on Lexical and Computational Semantics*. Association for Computational Linguistics, June 2013.
- [21] P. Koehn, "Statistical machine translation," <https://www.statmt.org/>, web download.