Learning Embeddings from Scientific Corpora using Lexical, Grammatical and Semantic Information

Andres Garcia-Silva
agarcia@expertsystem.com
Expert System
Madrid, Spain

Ronald Denaux
rdenaux@expertsystem.com
Expert System
Madrid, Spain

Jose Manuel Gomez-Perez
jmgomez@expertsystem.com
Expert System
Madrid, spain

ABSTRACT

Natural language processing can assist scientists to leverage the increasing amount of information contained in scientific bibliography. The current trend, based on deep learning and embeddings, uses representations at the (sub)word level that require large amounts of training data and neural architectures with millions of parameters to learn successful language models, like BERT. However, these representations may not be well suited for the scientific domain, where it is common to find complex terms, e.g. multi-word, with a domain-specific meaning in a very specific context. In this paper we propose an approach based on a linguistic analysis of the corpus using a knowledge graph to learn representations that can unambiguously capture such terms and their meaning. We learn embeddings from different linguistic annotations on the text and evaluate them through a classification task over the SciGraph taxonomy, showing that our representations outperform (sub)word-level approaches.

CCS CONCEPTS

• **Computing methodologies** → **Natural language processing**; **Neural networks**; *Semantic networks*; *Machine learning approaches*.

KEYWORDS

NLP, neural networks, convolutional neural networks, embeddings, text classification

1 INTRODUCTION

Nowadays scholarly communications are evolving, thanks to the effort of research communities, funding agencies and publishers, beyond the conventional delivery method based on documents to gain better visibility, reuse capabilities and to foster a broader data accessibility[8]. The list of enhancements is wide and include the availability of supporting material such as code [1] and research software[29], the use of Digital Object Identifiers to favor reusability and proper credit to authors, the emergence of specialized academic search engines such as semantic scholar, and the adoption of the FAIR principles [32] to make data findable, accessible, interoperable and reusable.

Aligned with the FAIR principles, in particular with the goal of assisting humans and machines in managing data, research objects [2, 3, 13, 33] encapsulate and annotate semantically all the resources involved in a research endeavour enabling data interoperability and machine-readability among other benefits. In addition, publishers have started releasing knowledge graphs such as Springer nature SciGraph[2], an open linked data graph about publications from the editorial group and cooperating partners, and the Literature Graph in Semantic scholar [1]. Nevertheless the knowledge of the scholar communications is still mainly text which is difficult to process by software agents. Research objects shed some light on the publication content with the semantic annotations, however they are user-generated and scarce in existing repositories [14]. Semantic scholar, on the other hand, uses Natural Language Processing to extract keywords and identify topics relevant for the publications.

In fact NLP technology is progressing at a fast pace thanks to word embeddings [19] and pre-trained language models based on transformers[30] that have allowed to improve the state of the art on different evaluation tasks [12, 24]. Most of existing word embeddings and pre-trained language models use sequences of characters, word pieces, and words in a sentence as their main input. However, in the scientific domain there are terms consisting of more than one word that have a domain-specific semantics. For example the meaning of a term such as *Molecularly imprinted polymer*[3] can be hardly identified from the single words, word pieces or other character-based representations, and hence the neural models used for NLP need to learn the relation between the single words, word pieces or characters, requiring complex architectures with a high number of parameters to optimize, and a huge amount of training data.

Scientific terminology is domain specific and scarce in a general corpus and hence accumulating the necessary amount of evidence from documents to identify it a as single entity with a specific meaning is very unlikely if we analyse single words and sub words representations. On the other hand, precisely in the scientific domain, the amount of structured resources, including catalogs, taxonomies and knowledge graphs with specific terminology and their corresponding definitions is available. Thus, the question raises, what is the minimum information unit or their combination thereof, which allows for efficient representations in vector form and at the same time can be linked to a semantically significant concept?

In this paper we propose to generate embeddings using surface forms, lemmas and concepts that are able to represent complex terms consisting of more than one word. The linguistic information is the result of applying a linguistic analysis that relies on a knowledge graph where linguistic knowledge is encoded. The linguistic analysis performs a grammatical, syntactical and semantic analysis to recognize and disambiguate terms that can consist of more than

[1] see the Data Citation initiative at https://doi.org/10.25490/a97f-egyk

[2] SciGraph homepage: https://www.springernature.com/gp/researchers/scigraph

[3] According to Wikipedia a Molecularly imprinted polymer is a polymer that has been processed using the molecular imprinting technique

one word. We generate embeddings from a scholarly communications corpus for single and joined representations (surface forms, lemmas, part-of-speech, and concepts). We experiment with these embeddings in a text classification task where the goal is to classify academic publications in a topic taxonomy.

Our results show that using linguistic annotation embeddings helps to learn better classifiers when compared to those learned only with words or subword embeddings. According to our experimentation the best approach is to use surface form and lemma embeddings jointly. When surface form and lemma embeddings are enriched with gramma information embeddings, like part-of-speech tag embeddings, the classifier with the greatest precision is learned. On the other hand, concept embeddings results were mixed probably due to the general-purpose annotator used in the experiments with a limited coverage of the scientific domain vocabulary.

This papers is structured as follows. Section 2 describes the related work and the paper contributions. Section 3 summarizes the approach to learn the embeddings for linguistic annotations. Next, Section 4 presents the experimental work where we evaluate the embeddings in a text classification task. Finally section 5 presents the conclusions and future lines of work.

2 RELATED WORK

Recent work in distributional representation of words has moved from static [6, 17, 19, 21, 28] to contextualized word embeddings [12, 22], in an effort to generate them dynamically according to the context and deal with phenomena like polysemy and homonymy. A main problem with traditional words embeddings is that unseen words or rare words are not represented in the distributional space and hence considered as out-of-vocabulary (OOV) words. To overcome the OOV problem different embedding representations have been proposed including character level used in ELMO [22], character-n-grams used in FastText [5], subwords used in GPT [23] and word pieces [27] used in BERT [12].

In parallel researchers have proposed to learn jointly concepts and word embeddings as an alternative approach to cope with the ambiguity of the language. For example Camacho-Collados et al. [9] relies on Wikipedia and Chen et al. [10] on WordNet to generate concept embeddings. Many approaches learn embeddings straight from knowledge graphs [7, 20, 25, 26], and others use linguistic annotations on a text corpus [11, 18].

In the scientific domain, Wang et al. [31] highlighted the limitations of general-purpose word embeddings in NLP tasks. So as to deal with such limitations Beltagy et al. [4] use BERT[12] to learn embeddings from the scientific domain. In this work adopt the Vecsigrafo approach [11] to generate embeddings from a scientific corpus for surface forms, lemmas and concepts. The vecsigrafo embeddings encodes linguistic information in contrast to approaches like Beltagy et al. [4] that relies on word pieces.

The main contribution of this paper is a comprehensive experimentation in the scientific domain with Vecsigrafo embeddings jointly learned from linguistic annotations and compare them with word and subword embeddings.

3 LEVERAGING LEXICAL, GRAMMATICAL AND SEMANTIC INFORMATION

To learn embeddings from different linguistic annotations we use Vecsigrafo [11], a method to learn embedding for linguistic annotations on a text corpus. Vecsigrafo extends the Swivel algorithm [28] to jointly learn embeddings for surface forms, lemmas, grammar types, and concept on a corpus enriched with linguistic annotations. Vecsigrafo embeddings outperformed the previous state of the art in word and word-sense embeddings by co-training surface form, lemma and concept embeddings as opposed to training each individually.

In contrast to simple tokens produced by space separation tokenization, linguistic annotations used in Vecsigrafo are based on terms that are related to one or more words. Surface forms are terms as they appear in the text, and lemmas are the base form of these terms. Source forms and lemmas can refer to concepts in a knowledge graph. For example, table 1 shows the linguistic annotations added to a text excerpt taken from a publication. Note how at the surface form level some tokens are grouped into terms like *local anesthetic* and *phrenic nerve*, and at the lemma level some surface forms such as *concerns* and *relating* are turned into their base form *concern* and *relate*. The grammar information indicates the role of the terms as nouns (N), verbs(V), noun and verb phrases (NP, VP), prepositions (P) and punctuation marks (PNT). In addition some of the terms are related to the concepts like like *local anesthetic* that is annotated with the concept *en%23107824862* that is defined as *An anesthetic that numbs a local area.*

Formally, Vecsigrafo generates, from a corpus an embedding space $\Phi = \{(x, e) : x \in SF \cup L \cup G \cup C, e \in \mathbf{R}^n\}$ where SF, L, G, and C are sets of surface forms, lemmas, grammar types, and concepts. One of the benefits of Vecisgrafo is that concept embeddings contribute to identifying the intended meaning of ambiguous terms in the corpus since the term and concept embeddings are learned jointly. To use Vecsigrafo embeddings in Φ we need to annotate the target corpus with the linguistic elements used to learn the embeddings. Note that embeddings representing linguistic annotations for the same term can be merged to generate a single embedding for the term, for example, by applying vector operations such as concatenation or averaging, or dimensional reduction techniques like PCA or SVD.

4 EXPERIMENTAL WORK

In this section we describe the scholarly communication corpus used to learn the linguistic embeddings, the NLP toolkit used to annotate the corpus, the neural network that uses the linguistic embeddings to classify the research publications, and report the evaluation results of the classifiers.

4.1 Embeddings for Scholarly Communications

SciGraph [15] is a linked open data platform for the scientific domain. It contains information from the complete research process: research projects, conferences, authors and publications, among others. The knowledge graph contains more than 1 billion facts about objects of interest to the scholarly domain, distributed over some 85 million entities described using 50 classes and more than 250 properties. Most of the knowledge graph is available under CC

Concept	en%2326973	en%2377696	-	en%23107824862		en%23100274160	-	en%23100737313		en%23101569578
Grammar	N	V	P	NP		N	PNT	NP		N
Lemma	concern	relate	to	local anesthetic		toxicity	,	phrenic nerve		blockade
Surface Form	concerns	relating	to	local anesthetic		toxicity	,	phrenic nerve		blockade
Token	concerns	relating	to	local	anesthetic	toxicity	,	phrenic	nerve	blockade

Table 1: Linguistic annotations and tokens generated for the text excerpt "concerns relating to local anesthetic toxicity, phrenic nerve blockade" extracted from an actual publication.

Linguistic annotations	Total	Distinct	Embeddings
Token	707M	1,486,848	1,486,848
Surface Form	805M	5,090,304	692,224
Lemma	508M	4,798,313	770,048
grammar	804M	25	8
Concept	425M	212,365	147,456

Table 2: Token and linguistic annotations, and embeddings generated from text in the title and abstract of research articles and book chapters published between 2001 to 2017 and available in Scigraph. The number of distinct linguistic annotations is different than the embeddings because we filter out articles and auxiliary verbs and apply a minimum frequency threshold.

BY 4.0 License (i.e., attribution) with the exception of abstracts and grant metadata, which are available under CC BY-NC 4.0 License (i.e., attribution and non-comercial) A core ontology expressed in OWL encodes the semantics of the data in the knowledge graph consisting of 47 classes and 253 properties. From SciGraph we extract publications including articles and book chapters published from 2001 to 2017. We use the titles and abstracts of the publications to generate the corpus with roughly 3.2 million publications, 1.4 million distinct words, and 700 million tokens.

Next we use Expert System NLP suit (Cogito) to parse the text and add linguistic annotations. Cogito disambiguator relies on its own knowledge graph called Sensigrafo, that encodes the linguistic knowledge in a way similar to WordNet, and applies a rule-based approach to disambiguation. The Sensigrafo contains about 400K, lemmas and 300K concepts interlinked via 61 relation types. Note that we could have used any other NLP toolkit as long as it generates the linguistic annotations used in this work. The corpus parsing and annotations generated by Cogito are reported in table 2.

For each linguistic element we learned an initial set of embeddings with 300 dimensions using Vecsigrafo. The difference between the number of learned embeddings and the linguistic annotations is due to a filter that we applied based on previous results [11]. We filter out elements with grammar type article, punctuation mark or auxiliary verbs and generalize tokens with grammar type entity or person proper noun, replacing the original token with special tokens *grammar#ENT* and *grammar#NPH* respectively. In addition, to these embeddings, we learned 10 Vecsigrafo embedding spaces for the possible combinations of size 2 and 3 between the linguistic elements *sf, l, g* and *c*.

Embeddings Generation	Precision	Recall	F-measure
Normal Distribution	0,7596	0,6775	0,7015△
Optimized by CNN	0,8062	0,767	0,7806▽

Table 3: Evaluation results for classifiers using token-based embeddings generated randomly and following the normal distribution (baseline) and optimized by the convolutional neural network (upper bound)

4.2 Evaluation Task

Publications in Scigraph have one or more *field of research codes* that classify the documents in 22 categories such as Mathematical Sciences, Engineering or Medical and Health Sciences. Thus, we can formulate a multi-label classification task that aims at predicting one or more of these 22 first level categories for each publication.

Embeddings are the natural numerical representation of text for neural networks. Kim [16] shows that Convolutional neural networks CNN were fitted for text classification and his results improved the state of the art on different text classification tasks and benchmarks. CNN are based on convolutional layers that slide filters (aka kernels) across the input data and return the dot products of the elements of the filter and each fragment of the input. These convolutions allows the network to learn features from the data, alleviating the manual selection required in traditional approaches. Stacking several convolutional layers allows feature composition, increasing the level of abstraction from the initial layers to the output.

To learn the classifier we use an off the shelf CNN implementation available in Keras, with 3 convolutional layers, 128 filters and a 5-element window size. As corpus we use 187795 articles available in SciGraph published in 2011. To evaluate the classifiers we use ten-fold cross-validation and precision, recall and f-measure as metrics. We use a vocabulary with maximum 20K entries, and sequences size 1000.

As baseline, we train a classifier that learns from embeddings generated randomly following a normal distribution. As upper bound we learn a classifier that is able to optimize the embeddings in the learning process. The evaluation of baseline and upper bound classifiers are presented in table 3.

4.3 Classifiers using vecsigrafo embeddings

We train classifiers using single Vecsigrafo embeddings for each linguistic annotation (*sf, l, c*) and for the ten 2, and 3 size combinations of (*sf, l, g, c*). Grammar embeddings were not evaluated

independently due to the low number of distinct grammar types used to annotate the terms. When using embeddings of two or three linguistic annotations two different approaches are used. The first approach relies on a single vocabulary containing at most 20K entries per each linguistic annotations in the text, and no merging operation is carried out, while in the second one embeddings are merged using concatenation or average. Evaluations results are reported in table 4.

4.4 Lemmas better than surface forms and tokens

Regarding single linguistic annotations, **lemma *l* and surface form *sf* embeddings contribute to learn the better classifier than using token *t* embeddings respectively**. This shows that the classifier learning process benefits from the conflation of different term and word variations (*sf*, *t*) into a base form (*l*). However, grouping raw tokens into terms (*sf*) only generates a slight improvement in the classifier performance with respect to using only tokens (*t*). On the other hand, concept (*c*) embeddings performance in this task is worst than *t* embeddings. The low number of *c* embeddings (see table 2) compared to the number of tokens and the other linguistic annotations affect negatively the learning process. The difference between concepts and tokens is consequence of limited coverage of the general-purpose annotator used in a highly specialized domain as the scientific.

4.5 Lemmas and surface forms the best combination

To analyse the results of the different combinations of embeddings for linguistic annotations we focus on each evaluation metric. Regarding precision the top 2 classifiers are learned from combinations of *sf*, *l* and *g*. In addition note that the common linguistic element in the top 6 classifiers is *g* combined either with *sf* or *l*, and in general removing *g* produced least precise classifiers. Thus, **precision-wise the part-of-speech information in combination with surface forms and lemmas is very relevant**. Semantic information (*c*) also contributes to enhance precision when it is combined with lemmas and surface forms, or with lemmas and grammar information. In addition, the precision of 16 classifiers out of 22 is better than the upper bound reported in table 3, where the embeddings are optimized in the classifier learning phase, even though vecsigrafo embeddings were not learned for this specific purpose.

The recall analysis shows a different picture since the grammar information (*g*) does not seem to have a decisive role on the classifier performance. **Surface forms and lemmas generates the classifier with highest recall**. Nevertheless, in this analysis concepts (*c*) gain more relevance always in combination with either *sf* or *l*. The combination of *l* and *c* seems to benefit recall since it is presented in 3 of the top 5 classifiers. In contrast, when concepts are combined with *sf* the recall is lower. In general *g*-based embedding combinations generate classifiers with lower recall. Note that none of the classifiers reached the recall of the upper bound classifier.

The f-measure data shows more heterogeneous results since by definition it is the harmonic mean of precision and recall, and hence the embedding combinations that generate the best f-measure needs a high precision and a high recall. **The combination of surface forms *sf* and lemmas *l* embeddings is at the top of the f-measure ranking**, followed by their combination with *c*. In general, concept embeddings improves the f-measure when combined with either lemmas or surface forms. However, when used in conjunction with lemmas and surface form embeddings the performance is worse. In general, due to the low coverage of concepts in the scientific domain the classifiers that relies only on *c* embeddings perform worst even when combined with grammar information. Similarly surface forms offer poor performance when combined with grammar information.

Finally note how the best classifiers were learned when the linguistic annotation embeddings are used independently which contrast to the worse results achieved when merging the embeddings.

Linguistic Annotations	Merging	Precision	Recall	F-Measure↓
sf_l	-	0,8104	***0,7638***	***0,7818***
sf_l_c	-	***0,8135***	***0,7598***	***0,7809***
l_c	-	0,8102	***0,7604***	***0,7797***
l_g_c	-	0,8099	***0,7592***	***0,7791***
sf_g_c	-	0.8126	0.7585	0.7790
sf_l	Avg	0,8093	0,7588	***0,7787***
l_g_c	Avg	***0,8125***	0,7558	0,7779
sf_l_g	-	***0,8144***	0,7549	0,7779
sf_l_c	Avg	0,8080	0,7581	0,7773
sf_g_c	Avg	0.8137	0.7548	0.7769
sf_l_g	Concat	***0,8148***	0,7543	0,7765
l_c	Avg	0,8040	***0,7592***	0,7763
sf_c	-	0,8096	0,7549	0,7754
l_g	-	0,8121	0,7498	0,7728
l	-	0,8035	0,7539	0,7728
sf_c	Avg	0,8023	0,7543	0,7722
l_g	Concat	0,8077	0,7472	0,7688
sf	-	0,8030	0,7477	0,7684
t	-	0,8008	0,7491	0,7679
sf_g	-	***0,8124***	0,7387	0,7653
c	-	0,7973	0,7453	0,7650
sf_g	Concat	0,8101	0,7317	0,7648
c_g	-	0,8095	0,7357	0,7629
c_g	Concat	0,8076	0,7320	0,7596

Table 4: Classifiers learned using vecsigrafo embeddings and token embeddings (in grey row) sorted descently by F-Measure. Only the best classifier for either average or concatenation merging operation is reported. Italic and Bold font indicate the top 5 results per metric. The top value per metric is underlined

4.6 Words and subwords

We also test embeddings generated from word constituents. We resorted to FastText[6] since Vecsigrafo approach was not designed to generate embeddings for word constituents. We use FastText to generate token and character-ngram embeddings, with n ranging from 3 to 6. We use these embeddings to learn the classifiers using the same CNN architecture and evaluation procedure used in the experiments described above. Evaluation results, presented in table

FastText Embeddings	Precision	Recall	F-Measure
t	0.8236	0.7493	0.7770
t + character-ngrams	0.8255	0.7429	0.7724

Table 5: Evaluations of a classifier learned character-ngrams generated with FastText.

5, shows that token embeddings are better than using token and character-ngram embeddings, which is in line with our assumption that using subword representations could be not convinient in the scientific domain. Note that one of the benefits of using character-ngram embeddings is to avoid the out of the vocabulary words (OOV). However, in our case, the embeddings were learned from the whole scigraph corpus so we do not face the OOV problem in our experiments.

On the other hand, results in table 4 and 5 are not directly comparable since the embeddings are generated with a different algorithms (FastText vs Vecsigrafo). For example FastText token embeddings generate a better classifier than using Vecisgrafo token embeddings, and remarkably FastText embeddings in both cases reach the highest precision of all the tested embeddings. Nevertheless, we can see that the f-measure of the classifier that uses FastText character-ngram embeddings is lesser than the first 11 results reported in table 4, including the classifier that uses only lemmas.

5 CONCLUSIONS

Natural language processing has the potential to help scientists to manage and get insights out of the huge amount of scholarly communications available. Nowadays deep learning techniques based on word embeddings and language models have advanced the state of the art in different NLP tasks. Nevertheless, the predominant approach in NLP is to use word or subword representations as the input of deep neural architectures that requires large corpora to learn performing language models. However, in contrast to general-purpose corpora the scientific vocabulary often contains complex terms comprising more than one word with the additional characteristic that these terms are very specific and only make sense in certain fields of knowledge (e.g., Cosmic Microwave Background Radiation). Thus models using word or subword representations could have problems to gather the necessary textual evidence to capture their meaning.

To overcome the word and subword representation limitation we propose to use embeddings based on linguistic annotations such as surface forms, lemmas, part-of-speech information, and concepts. These embeddings are jointly learned from a corpus of scientific communications using an existing approach called Vecsigrafo. We evaluate the linguistic annotation embeddings in a multilabel classification where the goal was to assign a scientific topic to each publication. Our evaluations results show that lemmas help to learn better classifiers than using space-separated words and subword representations based on character-ngrams. The best results were achieved when lemma and surface forms were used jointly. Grammar information was very useful for high precision. Concepts, on the other hand, were less helpful in general mainly due to the low coverage of concepts in the scientific domain. Since part of the analysis that identify surface forms and lemmas are based on lexical and syntactical analysis the coverage was higher.

As future work we want evaluate the linguistic annotation embeddings on other evaluation tasks different from text classification where understanding the the glossary can have more impact like entailment and question and answering. In addition, another line of research is to evaluate the impact of the linguistic annotations when used as input representation to learn language models.

ACKNOWLEDGMENTS

This research has been supported by The European Language Grid project funded by the European Unions Horizon 2020 research and innovation programme undergrant agreement No 825627 (ELG).

REFERENCES

[1] Waleed Ammar, Dirk Groeneveld, Chandra Bhagavatula, Iz Beltagy, Miles Crawford, Doug Downey, Jason Dunkelberger, Ahmed Elgohary, Sergey Feldman, Vu Ha, Rodney Kinney, Sebastian Kohlmeier, Kyle Lo, Tyler C. Murray, Hsu-Han Ooi, Matthew E. Peters, Joanna L. Power, Sam Skjonsberg, Lucy Lu Wang, Christopher Wilhelm, Zheng Yuan, Madeleine van Zuylen, and Oren Etzioni. 2018. Construction of the Literature Graph in Semantic Scholar. In *NAACL-HLT*.

[2] S Bechhofer, I Buchan, D De Roure, P Missier, J Ainsworth, J Bhagat, P Couch, D Cruickshank, M Delderfield, I Dunlop, M Gamble, D Michaelides, S Owen, D Newman, S Sufi, and C Goble. 2013. Why linked data is not enough for scientists. *Future Generation Computer Systems* 29, 2 (2013), 599 – 611. https://doi.org/10.1016/j.future.2011.08.004 Special section: Recent advances in e-Science.

[3] K Belhajjame, O Corcho, D Garijo, J Zhao, P Missier, DR Newman, R Palma, S Bechhofer, E Garcia-Cuesta, JM Gomez-Perez, G Klyne, K Page, M Roos, JE Ruiz, S Soiland-Reyes, L Verdes-Montenegro, D De Roure, and C Goble. [n. d.]. Workflow-Centric Research Objects: A First Class Citizen in the Scholarly Discourse. 1–12. http://ceur-ws.org/Vol-903/paper-01.pdf

[4] Iz Beltagy, Arman Cohan, and Kyle Lo. 2019. SciBERT: Pretrained Contextualized Embeddings for Scientific Text. arXiv:arXiv:1903.10676

[5] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching Word Vectors with Subword Information. *arXiv preprint arXiv:1607.04606* (2016).

[6] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* 5 (2017), 135–146.

[7] Antoine Bordes, Nicolas Usunier, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-Relational Data. *Advances in NIPS* 26 (2013), 2787–2795. https://doi.org/10.1007/s13398-014-0173-7.2 arXiv:arXiv:1011.1669v3

[8] Philip E. Bourne, Timothy W. Clark, Robert Dale, Anita de Waard, Ivan Herman, Eduard H. Hovy, and David Shotton. 2012. Improving The Future of Research Communications and e-Scholarship (Dagstuhl Perspectives Workshop 11331). *Dagstuhl Manifestos* 1, 1 (2012), 41–60. https://doi.org/10.4230/DagMan.1.1.41

[9] José Camacho-Collados, Mohammad Taher Pilehvar, and Roberto Navigli. 2016. NASARI: Integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities. *Artificial Intelligence* 240 (2016), 36–64. https://doi.org/10.1016/j.artint.2016.07.005

[10] Xinxiong Chen, Zhiyuan Liu, and Maosong Sun. 2014. A Unified Model for Word Sense Representation and Disambiguation. In *EMNLP*. 1025–1035.

[11] R Denaux and JM Gomez-Perez. 2019. Vecsigrafo: Corpus-based Word-Concept Embeddings-Bridging the Statistic-Symbolic Representational Gap in Natural Language Processing. *To appear in Semantic Web Journal* http://www.semantic-web-journal.net/system/files/swj2148.pdf (2019).

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805* (2018).

[13] Andres Garcia-Silva, Jose Manuel Gomez-Perez, Raul Palma, Marcin Krystek, Simone Mantovani, Federica Foglini, Valentina Grande, Francesco De Leo, Stefano Salvi, Elisa Trasatti, Vito Romaniello, Mirko Albani, Cristiano Silvagni, Rosemarie Leone, Fulvio Marelli, Sergio Albani, Michele Lazzarini, Hazel J. Napier, Helen M. Glaves, Timothy Aldridge, Charles Meertens, Fran Boler, Henry W. Loescher, Christine Laney, Melissa A. Genazzio, Daniel Crawl, and Ilkay Altintas. 2019. Enabling FAIR research in Earth Science through research objects. *Future Generation Computer Systems* 98 (2019), 550 – 564. https://doi.org/10.1016/j.future.2019.03.046

[14] Jose Manuel Gomez-Perez, Raul Palma, and Andres Garcia-Silva. 2017. Towards a human-machine scientific partnership based on semantically rich research objects. In *2017 IEEE 13th International Conference on e-Science (e-Science)*. IEEE, 266–275.

[15] Tony Hammond, Michele Pasin, and Evangelos Theodoridis. 2017. Data integration and disintegration: Managing Springer Nature SciGraph with SHACL and OWL.. In *International Semantic Web Conference (Posters, Demos and Industry Tracks) (CEUR Workshop Proceedings)*, Nadeschda Nikitina, Dezhao Song, Achille Fokoue, and Peter Haase (Eds.), Vol. 1963. CEUR-WS.org. http://dblp.uni-trier.de/db/conf/semweb/iswc2017p.html#HammondPT17

[16] Yoon Kim. 2014. Convolutional Neural Networks for Sentence Classification. In *EMNLP*.

[17] Omer Levy and Yoav Goldberg. 2014. Neural Word Embedding As Implicit Matrix Factorization. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2 (NIPS'14)*. MIT Press, Cambridge, MA, USA, 2177–2185. http://dl.acm.org/citation.cfm?id=2969033.2969070

[18] Massimiliano Mancini, José Camacho-Collados, Ignacio Iacobacci, and Roberto Navigli. 2017. Embedding Words and Senses Together via Joint Knowledge-Enhanced Training. In *CoNLL*.

[19] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013). arXiv:1301.3781 http://arxiv.org/abs/1301.3781

[20] Maximilian Nickel, Lorenzo Rosasco, and Tomaso A. Poggio. 2016. Holographic Embeddings of Knowledge Graphs. In *AAAI*.

[21] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation.. In *EMNLP*, Vol. 14. 1532–1543.

[22] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, 2227–2237. https://doi.org/10.18653/v1/N18-1202

[23] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf* (2018).

[24] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog* 1, 8 (2019).

[25] Petar Ristoski and Heiko Paulheim. 2016. RDF2Vec: RDF graph embeddings for data mining. In *International Semantic Web Conference*, Vol. 9981 LNCS. 498–514. https://doi.org/10.1007/978-3-319-46523-4_30

[26] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling Relational Data with Graph Convolutional Networks. arXiv:1703.06103

[27] Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 5149–5152.

[28] Noam Shazeer, Ryan Doherty, Colin Evans, and Chris Waterson. 2016. Swivel: Improving Embeddings by Noticing What's Missing. *arXiv preprint* (2016). arXiv:1602.02215

[29] Arfon M. Smith, Daniel S. Katz, and Kyle E. and Niemeyer. 2016. Software citation principles. *PeerJ Computer Science* 2 (Sept. 2016), e86. https://doi.org/10.7717/peerj-cs.86

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *CoRR* abs/1706.03762 (2017). arXiv:1706.03762 http://arxiv.org/abs/1706.03762

[31] Yanshan Wang, Sijia Liu, Naveed Afzal, Majid Rastegar-Mojarad, Liwei Wang, Feichen Shen, Paul Kingsbury, and Hongfang Liu. 2018. A comparison of word embeddings for the biomedical natural language processing. *Journal of Biomedical Informatics* 87 (2018), 12 – 20. https://doi.org/10.1016/j.jbi.2018.09.008

[32] Mark Wilkinson and et al. 2016. The FAIR Guiding Principles for scientific data management and stewardship. *Nature Scientific Data* 160018 (2016). http://www.nature.com/articles/sdata201618

[33] J Zhao, JM Gomez-Perez, K Belhajjame, G Klyne, E GarcÃŋa-Cuesta, A Garrido, KM Hettne, M Roos, D De Roure, and C Goble. 2012. Why workflows break - Understanding and combating decay in Taverna workflows.. In *8th IEEE International Conference on E-Science*. 1–9. https://doi.org/10.1109/eScience.2012.6404482