Semantic Search with Concept Vectors in the PubMed Central Biomedical Literature Dataset

Robert M. Horton[1], Jerry Lau[2], Katherine E. Gundling[3], John Mount[1], and John Mark Agosta[4]

[1] Win-Vector Labs, San Francisco, CA USA
[2] phactMI
[3] University of California, San Francisco USA
[4] San José State University, San Jose, CA USA

Abstract. Semantic embedding models use deep learning techniques to compute vector representations so that passages of text with similar meaning will be assigned similar vectors. Similarity between embedding vectors is widely used to search databases: the embedding for a query passage is used to search a database for similar passages. Here we present an approach to generate vectors representing general concepts in a semantic embedding space, constructed from the coefficients of linear classifiers. This representation reframes model scoring as vector search. We illustrate the concept vector idea by training a wide range of ML classifiers on biomedical literature in US National Library of Medicine in the PubMed Central open-access dataset. Using this rich source of biomedical text and associated metadata we show two general approaches for training ML models on this data; predicting section heading patterns, and predicting indexing terms (MeSH). We present pragmatic interpretable screening criteria to evaluate biases in these models such that potential users can reason about their suitability in particular contexts.

Keywords: semantic search · interpretable macine learning · concept representation.

1 Introduction

Search of text datasets using semantic embeddings uses vector similarity, exploiting fast approximate nearest neighbor techniques. This is usually done by example, where the query is the vector embedding of a representative passage of text. Here we demonstrate a method to construct a query vector for a general concept, rather than an example. We use a logistic regression classifier to recognize a target category, then construct a query vector from the coefficients of that classifier. We show the mathematical basis for this approach and demonstrate how it scales to large datasets using a large number of models trained on the PubMed Central collection of open-access biomedical research articles.

1.1 Coefficient unit vectors

Cosine similarity is a commonly used metric for comparing embedding vectors. It can be computed from the dot product of the two vectors and their magnitudes:

$$S_C(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|} \tag{1}$$

Embedding vectors are commonly normalized to unit length (so that $\|\mathbf{x}\| = 1$), making the cosine similarity equal to the dot product and saving some computation. Here we exclusively use embedding vectors of unit length.

Logistic regression finds the probability of a binary outcome y given a vector of observed features $\mathbf{x}$ using this relationship:

$$P(y \mid \mathbf{x}) = \sigma(\beta_0 + \boldsymbol{\beta} \cdot \mathbf{x}) \tag{2}$$

where β_0 is a scalar valued intercept, $\boldsymbol{\beta}$ is a vector of coefficients the same length as the feature vector and σ is the logistic sigmoid function $\sigma(t) = \frac{1}{1+e^{-t}}$.

We can re-write this equation to use a coefficient vector of unit length:

$$P(y \mid \mathbf{x}) = \sigma(\beta_0 + \|\boldsymbol{\beta}\| \boldsymbol{b} \cdot \mathbf{x}) \tag{3}$$

where $\boldsymbol{b}$ is the coefficient unit vector $\frac{\boldsymbol{\beta}}{\|\boldsymbol{\beta}\|}$, and the length of the original coefficient vector ($\|\boldsymbol{\beta}\|$) is a scaling factor.

If we train logistic regression models using embedding vectors as the only predictive features, the coefficient unit vector $\boldsymbol{b}$ is of the same dimension as the embeddings. This gives us the mechanical result that we can compute the cosine similarity between $\boldsymbol{b}$ and any feature vector by a simple dot product. It also means that $\boldsymbol{b}$ represents a point in the semantic embedding space.

1.2 Framing prediction as search

Equation 3 shows that predicted probability is monotonically related to the value of the dot product. This means that the cases with the highest predicted probabilities are simply those with the highest value of the dot product, which is vector search.

2 PubMed and PubMed Central (PMC) data

The mid-December 2023 baseline ‘oa_comm’ subset of open-access full-text documents was downloaded in XML format from the PMC bulk-download service [10]. Section headings and text for each paragraph were extracted using Python scripts and loaded into a Postgres database table (’paragraphs’) indexed by PubMed identifier (paid) and paragraph number. Titles were given the pseudo-section heading ‘Title’, and placed as paragraph 0 of each article. Embeddings were computed in Python using the SentenceTransformers model ’all-mpnet-base-v2’ and loaded into a pgvector vector(768) column of a separate table, also indexed by pmid and paragraph number. Other metadata, including journal information, year of publication, and MeSH term associations were extracted from the Pubmed 2023 annual baseline [11]

3 Mining labels from section headings

The path of section and subsection headings leading to a paragraph is a kind of meta-data about the text, which distinguishes our approach from methods like Snorkel [6] that rely on finding patterns directly in the text itself. We illustrate a process of iterative improvement to develop labels leading to a model with improved classifier performance on a hand-labelled custom test set.

SRD evaluation use case: Scientific Response Documents (SRDs) are responses by pharmaceutical companies to unsolicited inquiries from healthcare providers that can extend beyond the product labeling. To conform with FDA Guidance on Responding to Unsolicited Requests for Off-Label Information they must be non-promotional, evidence based, and scientifically balanced. PhactMI, a nonprofit collaboration of pharmaceutical company medical information (MI) leaders who oversee the MI departments that creates these documents, has developed best practices guidelines on creating SRDs [3]. as well as a rubric to quantitively assess how well an SRD adheres to these guidelines. Though the rubric was originally intended for human evaluation, the organization is also investigating the possibility of evaluating some aspects automatically. Several criteria can potentially be automated by text classifiers [5]. Here we focus on the requirement that SRDs based on clinical trials should contain information regarding adverse effects.

Table 1. Iterative refinement of section heading patterns. A dataset is generated for a given pattern and used to train a classifier. High-scoring paragraphs and their section headings are studied to inspire modifications to the pattern, and the process is repeated. Patterns were matched in PostgreSQL, and models were fitted with glmnet.cv in R.

Regex Pattern	***AUC***
`adverse event`	0.837
`adverse.*(event|effect)`	0.846
`results.*adverse.*(event|effect)`	0.859
`results.*(adverse (event|effect)|tolerability|safety)`	0.864

Iterative pattern improvement process: We use a custom training set for each label, consisting of all the paragraphs from articles having a section matching the pattern. Evaluation is done on a common test set consisting of hand-labelled sentences from SRDs [5]. The model developed in that study had an AUC of 0.84, using expert labelling with Prodigy. Several of the models in Table 1 outperform that baseline, but we do not select patterns on test set performance alone. Focusing on cases with high scores from one pattern but not another helps our experts judge which variant is preferable, even if they have similar AUCs.

4 Predicting MeSH terms

Medical Subject Heading (MeSH) descriptors are indexing terms that capture the judgement of medical informaticists about which concepts from a defined vocabulary apply to a given article. MeSH terms are applied to articles, but we are predicting them for individual paragraphs. This is an example of multiple instance learning [2], [1] where we label each paragraph from an article with its MeSH terms, and predict as well as we can. We trained paragraph level predictive models on the 1000 most common MeSH terms in our dataset using LogisticRegressionCV in Python. We find that in our current application the multiple instance effect is not too damaging; fitting paragraph level logistic regression models using document labels gives models with good ordering statistics (e.g, area under the ROC curve), and only slightly affects probability calibration at the paragraph level.

4.1 Using predictions of new MeSH terms in older literature

From a set of MeSH terms added in 2022 [9] we selected 14 that fell within the domain of interest of our medical subject matter expert, and used models trained on these terms to search articles published in 2015. Several insights emerged:
Health Inequities: "Stronger scores are associated with excellent matches to the concept, enabling the researcher to home right in on the query of interest."
Brain-Gut Axis: "The highest-scoring paragraphs tended to be on topic. Many paragraphs were found with more moderate scores, and some of this content contains discussion of either brain or gut, but not both."
mRNA Vaccines: "The model retrieves a host of clinical trials and COVID-related data. The basic science-centered definition of this MeSH term appears to have been overwhelmed by pandemic-related realities."

The first example works as expected, while the other led us to hypothesize two failure modes: an inability to represent interactions between subconcepts in *Brain-Gut Axis*, and strong bias in the training data for *mRNA Vaccines*.

5 Interpretable evaluation criteria

Here we describe two approaches to qualitative evaluation of models based on interpretable criteria that can easily be applied to large numbers of models. These approaches can help weed out models with biases that seem inappropriate for a particular application. Detailed results of these evaluations can be found on our Github repository[4].

5.1 Clustering concept vectors

Concept vectors from 1014 MeSH classifiers were clustered hierarchically using the cosine distance metric and Ward's method [13] for agglomeration. The hierarchy tree was sliced at a sequence of smaller and smaller thresholds, partitioning the concept vectors into sets of increasingly focused clusters. This generated a dataframe where each concept vector is a row, and columns indicate the clusters at the different levels. The rows were sorted by the cluster assignments in order from larget to smaller clusters, resulting in similar rows being close together. We have also included columns containing the official definition for each term as well as its mtree strings showing the terms position in the 16-dimensional knowledge graph hierarchy [8]. These results are in the file 'concept_clusters.xlsx' in the 'understanding classifiers' directory of our Github repository.

Clustering helps to reveal biases reflecting the distributional peculiarities of the PMC training data. For example, the nearest neighbor of *Exercise* is *Sedentary Behavior*, which is basically the opposite of exercise. These terms have similar vectors because the concepts tend to co-occur; in other words, paragraphs about exercise are also likely to be about sedentary behavior because studies often compare people who exercise to those who do not.

Most of the concept vectors appear to fall into reasonable clusters from a biomedical perspective. For example, *Software* falls into a tight cluster with *Databases, Genetic* and *User-Computer Interface*, which is a close sibling to a tight cluster containing *Genome*, *Genome-Human*, and *Molecular Sequence Annotation*. As long as you are interested in software related to these domains, this concept vector may be reasonable, but for software in general it may not.

Some of these biases result in concept vectors that are much more specialized than the MeSH term alone might suggest. For example the vector for *Animals* is very tightly clustered with *Mice*, and more broadly clustered with terms reflecting the bias of biomedical literature toward animals as experimental subjects. These relationships are not implied in the definition of the *Animals* MeSH term [7]. But if you are interested in searching datasets containing passages about household pets, livestock, or exotic zoological specimens, the clustering analysis indicates that this concept vector is probably not appropriate.

5.2 Concepts versus definitions

The extensive MeSH documentation makes it possible to apply another general approach to characterizing our panel of 1014 MeSH concept vectors; we use each of them to score all 30598 definitions in a MeSH term dictionary by similarity, then rank the definitions by this score for each concept. We can summarize these results by the rank each model gives to its own definition. For 14.2% of our concept vectors the definition with the highest score is the one for the term the model was trained to recognize, and over 50% of the vectors ranked their own definition in the top 12 [12]. This simple analysis identifies a substantial fraction of the concept vectors that are not well described by the corresponding definition. Such mismatches could be due to biases in the training data (as described above for *Animals*), or inadequate definitions (the most extreme examples being terms for which no definitions are provides, which include *Male* and *Female*). It may require domain expertise to characterize these biases, or to construct better definitions. However, the good news is that we have trained and documented a large number of concept vectors that do in fact appear to target the MeSH term as described by its definition.

6 Conclusions and future directions

Concept vectors are points in an embedding space. We have shown how to compute them by fitting linear models and extracting coefficients. These vectors have meaning to the extent that points in the space have meaning. Embedding both documents and concepts in the same vector space allows fast retrieval both from concepts to instances (finding documents about a concept) and from instances to concepts (finding the concepts in a document).

PubMed Central is a vast data resource for training interpretable text classifiers. We have shown that matching patterns to section headings is a fast and flexible way to generate training data.

We can generate large numbers of concept vectors by predicting MeSH terms. Sometimes the concept vectors may not mean exactly what we expect from the labels, but by pooling loosely meaningful clues, relationships among points in the semantic space (clusters, for example) can provide interpretable evidence about their meanings. Fitting linear classifiers on a GPU should let us process datasets large enough to contain even relatively rare concepts. A larger collection of vectors will give us richer and more detailed associations.

PubMed contains tens of thousands MeSH definitions specifically designed to be interpretable by humans. They can be placed in the semantic embedding space along with corpus text and concept vectors. Relationships to definitions provide detailed clues about the meaning and biases of concept vectors.

Differences between what we expect from a label and what we get from a model reflect biases in the training set. We are developing more sophisticated interactive tools to help domain experts more easily explore and interpret this type of data to better understand the biases. Additionally, we are investigating approaches to extend linear models to take interactions and non-linearities into account.

As a form of transfer learning, the success of using these vectors in other fields will depend on alignment between the biases in the target corpus with those in the PMC training corpus. By providing interpretable evidence that reveals the biases of the concept vectors we have trained on PMC, we make it easier for developers of biomedical search- and retrieval-based applications to judge whether particular concept vectors are likely to carry the intended meaning over into their target corpus.

Acknowledgments. This study was conducted entirely at the authors' expense.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Babenko, B.: Multiple instance learning: Algorithms and applications (01 2008)
2. Dietterich, T.G., Lathrop, R.H., Lozano-Pérez, T.: Solving the multiple instance problem with axis-parallel rectangles. Artificial intelligence **89**(1-2), 31–71 (1997)
3. Hermes-DeSantis, E.R., Johnson, R.M., Redlich, A., Patel, B., Flanigan-Minnick, A., Wnorowski, S., Cortes, M.M., Han, C.W., Vine, E., Sarwar, H., Haydar, R., Jamil, A., Huang, T., Sandhu, S.K., Reilly, P.: Proposed Best Practice Guidelines for Scientific Response Documents: A Consensus Statement from phactMI. Ther Innov Regul Sci **54**(6), 1303–1311 (Nov 2020)
4. Horton, R., Mount, J., Agosta, J.M.: Pmc classifiers (github repository), `https://github.com/rmhorton/PMC_classifiers`, [Online; accessed 20-June-2024]
5. Lau, J., Bisht, S., Horton, R., Crisan, A., Jones, J., Inchiosa, M.E., Gantotti, S., Hermes-DeSantis, E.R.: Evolution of artificial intelligence in the pharmaceutical industry - gauging a potential use for creation of scientific response documents for addressing product medical information inquiries. Submitted (2024)
6. Ratner, A., Bach, S.H., Ehrenberg, H., Fries, J., Wu, S., Ré, C.: Snorkel: rapid training data creation with weak supervision (2019)
7. US National Library of Medicine: Mesh term search results for 'animals', `https://www.ncbi.nlm.nih.gov/mesh/?term=animals`, [Online; accessed 20-June-2024]
8. US National Library of Medicine: Mesh tree structures, `https://www.nlm.nih.gov/mesh/intro_trees.html`, [Online; accessed 20-June-2024]
9. US National Library of Medicine: New mesh headings by subcategory, report for year 2022, `https://www.nlm.nih.gov/mesh/2022/download/2022NewheadingsbycategorywithScopeNotes.pdf`, [Online; accessed 18-June-2024]
10. US National Library of Medicine: Pmc open access subset ftp service, `https://ftp.ncbi.nlm.nih.gov/pub/pmc/oa_bulk/oa_comm/xml/`, [Online; accessed January 2024]
11. US National Library of Medicine: Pubmed download pubmed data annual baseline, `https://pubmed.ncbi.nlm.nih.gov/download/`, [Online; accessed February 2024]
12. US National Library of Medicine: Supplemental information: Rankings of definitions by concept vector, `https://github.com/rmhorton/PMC_classifiers/blob/main/understanding_classifiers/top_20_definition_terms.xlsx`, [Online; accessed 19-June-2024]
13. Wikipedia contributors: Ward's method (2024), `https://en.wikipedia.org/wiki/Ward%27s_method`, [Online; accessed 19-June-2024]