CONTENTS

Abstract

Medical Subject Headings (MeSH) indexing, which is to assign a set of MeSH main headings to citations, is crucial for many important tasks in biomedical text mining and information retrieval. Large-scale MeSH indexing has two challenging aspects: the citation side and MeSH side. For the citation side, all existing methods, including Medical Text Indexer (MTI) by National Library of Medicine and the state-of-the-art method, MeSHLabeler, deal with text by bag-of-words, which cannot capture semantic and context-dependent information well. We aim to implement RNNs (LSTMs, GRUs, RNNs) that incorporates deep semantic information for large-scale MeSH indexing. Recurrent Neural Networks (RNNs) have recently received much attention due to their efficacy in handling sequential data and we aim to capture this dependency in citations and predict the MeSH terms.

CHAPTER I

INTRODUCTION

1.1 Background

Medical Subject Headings (MeSH) is a comprehensive controlled vocabulary, which has been developed and maintained by National Library of Medicine (NLM), resulting in already 27 455 MeSH main headings (MHs) by 2015. One important usage of MeSH is to index citations in MEDLINE (NCBI Resource Coordinators, 2015; Nelson et al., 2004), to catalog documents, books as well as audiovisuals recorded in NLM. Currently one citation in MEDLINE is indexed by approximately 13 MHs on average. MeSH has also been used in many other applications in biomedical text mining and information retrieval, such as query expansion (Lu et al., 2010; Stokes et al., 2010), document clustering (Gu et al., 2013; Huang et al., 2011b; Zhu et al., 2009a,b) and document searching (Peng et al., 2015). Thus accurate MeSH indexing of biomedical documents is crucial for the biomedical researchers in formulating novel scientific hypothesis and discovering new knowledge.

Currently, human curators in NLM assign most relevant MeSH headings to documents, resulting in that 806,326 MEDLINE citations were indexed in 2015 (http://www.nlm.nih.gov/bsd/bsd_key.html). This work is very precious but clearly laborious, since to index an article, curators need to review the full text of the corresponding MEDLINE article, which is time consuming and prohibitively expensive For example, the average cost of annotating one MEDLINE article is estimated to be around $9.4 (Mork et al., 2013), meaning a huge total cost for indexing around one million documents per year. Also MEDLINE is rapidly growing, it would be more challenging for manual annotation to index all coming documents on time. To address this problem, NLM has developed an automatic MeSH indexing software, Medical Text Indexer (MTI), to assist MeSH curators. MTI recommends suitable MHs to each MEDLINE citation using the title and abstract as input (Aronson et al., 2004; Mork et al., 2014). MTI consists of two main components: MetaMap Indexing (MMI) and PubMed Related Citations (PRC). MMI extracts biomedical concepts from title and abstract, and then map them to corresponding MHs, while PRC attempts to find similar MEDLINE citations using a modified k-nearest neighbor (KNN) algorithm, PubMed Related Articles (PRA) (Lin and

Wilbur, 2007). The MHs of these similar citations are then extracted and combined with the MHs by MMI. After some post-processing steps, such as applying indexing rules, a ranked list of MHs is recommended to the MeSH indexers. From a machine learning viewpoint, automatic MeSH indexing can be considered as a large-scale multi-label classification problem (Liu et al., 2015), where each MH is a class label and each citation (instance) has multiple MHs. To address this multi-label classification problem, there are two main challenging aspects on the MeSH (label) and citation (instance) sides. First, on the MeSH (label) side, a large number of MHs have a highly biased distribution. For example, out of all 27,455 MHs, the most frequent MH, 'Humans', appears more than eight million times in the whole MEDLINE citations with abstracts, while the 20 000th frequent MH, 'Hypnosis, Anesthetic', appears only around 200 times. In addition, the number of annotated MHs for each citation varies greatly, ranging from more than 30 to less than 5. These aspects make the problem very challenging to estimate an effective and efficient prediction model for multi-label classification. Second, on the citation (instance) side, complicated semantics of biomedical documents cannot be effectively captured by a simple bag of words (BOW) approach, because a huge number of domain phrases, concepts and abbreviations exist in the biomedical literature. For example, similar concepts can be represented by different words, while the same word may have very different meanings depending upon the contexts. More concretely, 'malignancy', 'tumor' and 'cancer' are all very close concepts to each other, while 'CAT' can represent different genes, depending on organisms (Chen et al., 2004). Similarly, the same abbreviation is used as totally different concepts occasionally. For example, 'CCC' stands for Continuous Curvilinear Capsulorhexis in one citation (PMID:25291748), but Continuous Circular Course in another citation (PMID:23618326). However, simple BOW representation ignores the order of words and can hardly capture word semantics. In fact, using BOW, it would be very hard to distinguish different concepts represented by the same word, and also difficult to build connections between two different words representing similar concepts. Thus similar citations based on BOW representation may have totally different MHs. Table 1 gives a typical example, where a citation of interest is PMID:25236620, an article about cytopathology fellowship. Surprisingly, if one uses BOW, the most similar citation to PMID:25236620 among three articles in Table 1 becomes PMID:23416813, which is about the diagnosis of adult orbital masse by different techniques, although these two citations share only one MH, 'Humans'. The reason that this inaccurate similarity between two citations exists is: the term 'cytopathology' appears frequently in PMID:25236620 and also 'cytopathologically' and 'cytopathological' appear many times in PMID:23416813. These three terms have the same stemmed form, causing them to be regarded as the same term, and therefore leading to a very high similarity to these two citations in terms of BOW. Many studies have been carried out to tackle the challenging problem of automatic MeSH indexing based on different principles, such as k-nearest neighbor (KNN) (Trieschnigg et al., 2009), Naive Bayes (Jimeno-Yepes et al., 2012b), support vector machine (SVM) (Jimeno-Yepes et al., 2012a), Learning to Rank (LTR) (Huang et al., 2011a; Liu et al., 2015; Mao and Lu, 2013), deep learning (Jimeno- Yepes et al., 2014; Rios and Kavuluru, 2015) and multi-label learning (Liu et al., 2015; Tsoumakas et al., 2013). MeSHLabeler is a state-of-the-art automatic MeSH indexing algorithm, which won the first place in the large-scale MeSH indexing task of both BioASQ2 and BioASQ3 competition (http://bioasq.org) (Liu et al., 2015; Tsatsaronis et al., 2015). To address the distribution bias problem on the MH side, MeSHLabeler improves the performance of indexing MeSH by using a large number of different types of evidence regarding MH. These evidences are nicely integrated by using the framework of LTR. However, MeSHLabeler as well as other cutting-edge methods have not considered the problem on the citation (instance) side, and even MeSHLabeler still uses classic BOW representations, such as unigram and bigram. Recently, from the context of machine learning, the concept of dense semantic representation, such as Word2Vec (W2V), Word2Phrase (W2P) and Document2Vec (D2V), has been proposed to capture semantic and context information of text (Bengio et al., 2003; Le and Mikolov, 2014; Mikolov et al., 2013; Mitchell and Lapata, 2010; Socher et al., 2012, 2013). This new concept brings an opportunity to improve the performance of automatic MeSH indexing from the citation side. Specifically, we have developed DeepMeSH to address the largescale MeSH indexing problem. Instead of using rather shallow BOW representation, DeepMeSH incorporates deep semantic representation into MeSHLabeler to improve the performance of automatic indexing MeSH over large-scale document data. In particular, DeepMeSH uses a new dense semantic representation, D2V-TFIDF, which has both features of 'document to vector' (D2V) and 'term frequency with inverse document frequency' (TFIDF), meaning that D2V-TFIDF is more effective than individual D2V and TFIDF to find similar MEDLINE documents. Again Table 1 shows a typical result of using D2V-TFIDF. Regarding the citation in question, PMID:25236620, if we use dense semantic representation (DSR) only, PMID:23597252 can be selected as a highly similar citation, while if we consider both DSR and BOW (which is equivalent to D2V-TFIDF), another citation PMID:24576024 is more similar to PMID:25236620 than the other two citations. Importantly, this result is consistent with the similarity computed by using MHs only, as shown in the last column of Table 1. PMID:24576024 has the largest number of common MHs with PMID:25236620 among three articles in Table 1. Another point is that we use not only simple but rather diverse evidence in terms of dense semantic representation, following the framework of MeSHLabeler. That is, DeepMeSH takes advantage of new dense semantic representation to address the problem of the instance side and the MeSHLabeler framework to address the challenge on the label side. We validated the performance advantage of DeepMeSH by using BioASQ3 benchmark data with 6000 citations. DeepMeSH achieved the Micro Fmeasure of 0.6323, which is around 12% higher than that of 0.5637 by MTI and 2% higher than that of 0.6218 by MeSHLabeler.

Text classification is a foundational task in many NLP applications. Traditional text classifiers often rely on many human-designed features, such as dictionaries, knowledge bases and special tree kernels. In contrast to traditional methods, we introduce a recurrent neural network for text classification without human-designed features. In our model, we apply a recurrent structure to capture contextual information as far as possible when learning word representations, which may introduce considerably less noise compared to traditional window-based neural networks. However, overfitting is a serious problem in such networks with a large number of parameters.

Recurrent neural networks (RNNs) have been widely studied and used for various machine learning tasks which involve sequence modeling, especially when the input and output have variable lengths. Recent studies have revealed that RNNs using gating units can achieve promising results in both classification and generation tasks.

Although RNNs can theoretically capture any long-term dependency in an input sequence, it is well-known to be difficult to train an RNN to actually do so. One of the most successful and promising approaches to solve this issue is by modifying the RNN architecture e.g., by using a gated activation function, instead of the usual state-to-state transition function composing an affine transformation and a point-wise nonlinearity. A gated activation function, such as the long short-term memory (LSTM, Hochreiter & Schmidhuber, 1997) and the gated recurrent unit (GRU, Cho et al., 2014), is designed to have more persistent memory so that it can capture long-term dependencies more easily.

Deep neural networks contain multiple non-linear hidden layers and this makes them very expressive models that can learn very complicated relationships between their inputs and outputs. With limited training data, however, many of these complicated relationships will be the result of sampling noise, so they will exist in the training set but not in real test data even if it is drawn from the same distribution. This leads to overfitting and many methods have been developed for reducing it. These include stopping the training as soon as performance on a validation set starts to get worse, introducing weight penalties of various kinds such as L1 and L2 regularization and soft weight sharing (Nowlan and Hinton, 1992).

One such method of regularization in Deep Neural Networks is Dropout which prevents overfitting and provides a way of approximately combining exponentially many different neural network architectures efficiently. The term “dropout” refers to dropping out units (hidden and visible) in a neural network. Dropping a unit out means temporarily removing it from the network, along with all its incoming and outgoing connections. The choice of which units to drop is random.

CHAPTER II

RELATED WORK

Many studies for the problem of automatic MeSH indexing have used a relatively small- or middle-sized training data, or focus on only a small number of MHs. For example, NLM researchers explored the performance of several different machine learning algorithms, such as SVM, naive Bayes and AdaBoost, over a dataset of only around 300 000 citations (Jimeno-Yepes et al., 2012b, 2013). A clear limitation of these studies is that their approaches cannot be generalized to large-scale MeSH indexing in practice. The BioASQ challenge provides a more realistic and practical benchmark to advance the design of effective algorithms for largescale MeSH indexing (Tsatsaronis et al., 2015). Many effective algorithms have emerged through the BioASQ challenge, such as MetaLabeler (Tsoumakas et al., 2013), L2R (Huang et al., 2011a; Mao and Lu, 2013) and MeSHLabeler (Liu et al., 2015). However, all of them use the traditional shallow BOW representation. This is inadequate for capturing the semantic and context information of MEDLINE citations precisely, and therefore limits the performance of these models. Recently to address the problem of BOW representation, dense semantic representation for texts has been proposed in the machine learning domain (Bengio et al., 2003; Le and Mikolov, 2014; Mikolov et al., 2013). The performance of dense representation has, however, not yet been examined well in large-scale MeSH indexing, with one exception, in which weighted 'word to vector' (W2V) for MeSH indexing was explored (Kosmopoulos et al., 2015). The approach by (Kosmopoulos et al., 2015) is however rather primitive and not thorough enough to build a totally new approach for large-scale MeSH indexing. That is, they first use KNN to find similar citations using a new semantic representation and then the citations with high precision are just added to the results of MTI, meaning a kind of addition to MTI. Slight improvement sheds light on the possibility of exploring more effective representation for citations and developing efficient methods for integrating such representation to improve the performance of large-scale MeSH indexing.

CHAPTER III

METHODS

3.1 Overview

The MeSH indexing problem is to assign a certain number of MHs from the whole MHs list, which contains more than 27 000 terms, to a new MEDLINE citation. MeSHLabeler solves this problem by integrating multiple types of evidence generated from BOW representation in the framework of LTR. In contrast, by keeping the same, efficient framework of LTR, we integrate another type of strong evidence generated from dense semantic representation. Specifically, for each citation, it first generates a dense semantic vector, We train a LSTM classifier of MHs. These trained models are finally used to recommend suitable MHs in the framework.

3.2 Preliminary background

3.2.1 Data Handling

We start with 30,000 samples and we keep Pickling the data after each processing to ignore memory issues. All punctuations, apostrophes, brackets etc were removed from data. Sorted version of word counts were saved and the words were indexed according to this. We get our vocabulary of words used in such citations from here. One hot indices were generated from text with the help of the word indices. Simultaneously, a word to vec model was trained with our training data with vector size to be 200 and the window size as 10. All words were retained. A word-vector dictionary with all the words as key and their trained vectors as their values were pickled for future use. On the other hand, the MeSH label file contains 27,000 MHs and some of them grouped to one label. These groups were then extracted and one label at one time were given the same index but as a separate entity. A dictionary was maintained with the index as the key and the word as the value. One problem which was recognized later while testing was that many labels labelled in sample data was not present in the MeSH Label main file. These labels were then dropped and if the sample contained all words which were unrecognizable, then this sample was not taken into account of training. Using Keras, layers of a neural network was defined. The first being the embedding layer whose weights were initialized by the weights from the word to vector model. The output dimension for this layer is then of-course, 200 which is the word dimension. The layer is then followed by a layer of LSTM/GRU/SimpleRNN. Dropout was added for regularization and then the last layer was the output layer which outputs the probability. The weights of the trained model was saved for future use and updation.

Citation representation:

In RNN hidden unit representation, each citation can be represented with a vector consisting of all terms in a controlled vocabulary. Term frequency-inverse document frequency (TFIDF) is the most widely used scheme to weight each term. TF is the term frequency in a document and IDF is the inverse document frequency in the corpus. The idea behind TFIDF is that terms that occur more frequently in a particular document and also occur more in a subset of documents only should be emphasized more. However, our primary focus is to capture semantic relations in each citation and then predict the MH. Hence, we use the last LSTM output which contains a dense semantic representation for the citation. In this representation, the ordering of words when appearing in the document is kept.

Also document embedding can be generated directly from word or phrase embedding results. For example, we can compute the average of the word embedding of each word in a document, as the document embedding.

3.3 DeepMeSH

Both W2V and D2V are trained by using neural network with one hidden layer based on stochastic gradient descent and back-propagation. The procedure of generating these representations for a given citation is:

(i) we have to first generate vectors of W2V,

(ii) store the vectors of each word in an embedding vector.

(iii) initialize the embeddding layer of keras with the embedding vector above and train the LSTM layer for the D2V.

3.3.2 Using regular MeSH indexing methods with D2V

D2V is a dense semantic representation generated by considering semantic and context information in text, meaning that D2V can find semantic similar citations even without shared words. This would be helpful for identifying general MHs that are semantically related to many citations.

1. Classification using D2V :

    We can train a classifier for MH labels using D2V features.

CHAPTER IV

EXPERIMENTS

4.1 Data

We downloaded 23,343,329 citations of MEDLINE/PubMed from NLM, before the BioASQ challenge. 30,000 indexed citations with both abstracts and titles were locally stored as training data. For generating W2V features, we used Gensim's Word2Vec to preprocess MEDLINE raw text (Jiang and Zhai, 2007). A window dependency of 10 was chosen. No feature was removed. Each citation was then represented by a very dense vector of 200 dimension with the D2V weighting scheme.

4.2 Implementation and parameter setting

We used gensim (http://radimrehurek.com/gensim/models/word2vec.html) for the implementation of W2V. We first transformed all text into lowercase. The continuous bag of Words (CBOW) mode was then used to generate dense semantic representation. The dimensions of all dense vectors were set to 200. Learning the LSTM classifier for MHs required around 3 days. The maximum length of each citation was 6727. The dimension of each citation vector was varied from 50 to 150 with a better output accuracy to be noted with increasing dimension size. Increasing the dimension increased the overall training time. The dropout probability added was 0.5 which gives good results in most cases. The loss function taken was categorical cross-entropy with optimizer to be rmsprop and accuracy as a metric. While training the batchsize was 64 and validation split 0.2.

The total number of labels were 27883.

Training samples are as follows :

['From the above it is seen that the U. S. Public Health Service Research Grants program represents a sincere and continuing effort to supply Federal funds for the support of necessary additional research in the fields of medical and related sciences without interposing any degree of government restriction, control, supervision, or regimentation. The program is a scientific one, scientific guidance of which lies wholly in the hands of scientists.'

'1. T antigens of group A hemolytic streptococci have been obtained in soluble form by digestion of the bacterial cells with pepsin or trypsin. Large quantities of this antigen were readily extracted from type 1 strains, whereas only small amounts could be obtained from strains of other types. 2. The T antigen, prepared in this way from a type 1 strain, was partially purified by chemical precipitation and further enzymatic digestion. An active fraction, apparently protein in nature, was separated electrophoretically at pH 7.00. The separated material, pooled and analyzed at the same pH, gave only a single peak. The isoelectric point of this substance was about pH 4.50. An elementary analysis was obtained. Although the T antigen was resistant to digestion with proteolytic enzymes and ribonuclease, it was readily inactivated by heat, especially in acid media and in strong salt solutions. The serological activity of this purified T substance was lost after exposure to ultraviolet radiation. 3. Analysis by means of the ultracentrifuge showed that the material was polydisperse and therefore probably impure. 4. The soluble form of the T substance was active in the precipitin reaction, in the fixation of complement, in inhibition of T agglutination, and as an antigen when injected into rabbits. The antibodies produced did not protect mice against infection with virulent strains of hemolytic streptococci containing the same T antigen. 5. The immunological specificity of T antigen in soluble form is the same as that of the T antigen in the intact streptococcus from which it was derived.'

Their corresponding **one-hot vectors :**

[[0 0 0 ..., 3 8 21] [0 0 0 ..., 2 13 22] [0 0 0 ..., 64 28 36]

10 example samples with **indexed MeSH** :

[[3129], [1722, 1843, 13002, 24747, 24749], [5861], [11848, 13721, 15193], [18525], [18525], [21186], [18713], [8577, 23011], [12778]]

Their corresponding **labels** are :

[[Biomedical Research][Viral, Tumor, Mucosal, Streptococcal Infections, Streptococcus][Complement System Proteins], [Hemolysis, Ions, Magnesium][Orthomyxoviridae][Orthomyxoviridae], [Proteins], [O-Demethylating], [Western Equine, Rodentia], [Renovascular]

Results of **word2vec model**:

Similarity between the words 'convulsions' and ‘influenza' was given by 0.145.

Similarity between the words ‘smokers’ and ‘quitters’ was given by 0.462.

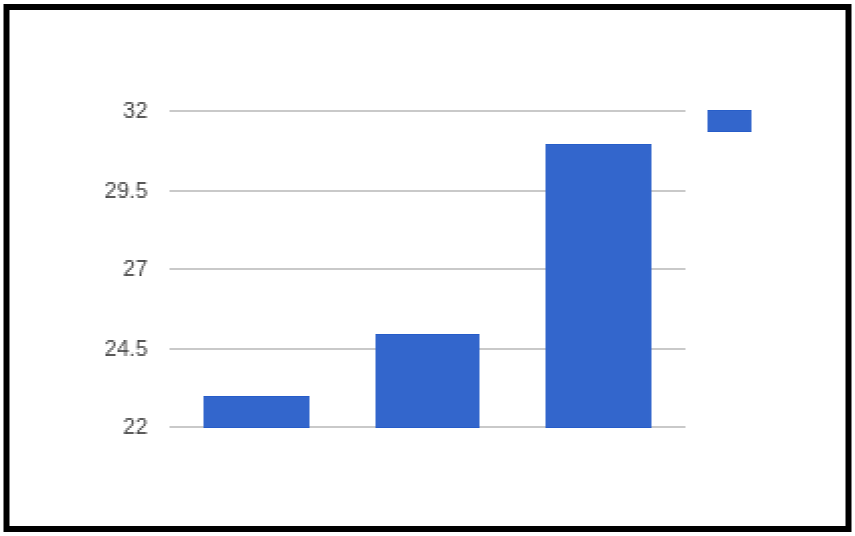
Now, looking at our **model outputs** carefully :

Noting that these results have been run on 30,000 samples each :

Model RNN Used	Accuracy
SimpleRNN	23%
GRU	25%
LSTM	31%

Our model performs much better than a randomized output (accuracy is much better than 1/27000 = .003704%)

Following are the **model vs accuracy(%)** graph :



Models vs accuracy (%) [SimpleRNN , GRU , LSTM]

Output of the model :

Graded numbers of marrow cells and 5 x 10(7) thymocytes were mixed in vitro and transplanted into X-irradiated (C3H x C57BL\/10)F(1) mice. Upon injection of sheep or chicken erythrocytes, splenic plaque-forming cells secreting IgM (direct PFC) or IgG (indirect PFC) hemolytic antibody were enumerated at the time of peak responses. Anti-sheep and anti-chicken primary PFC responses elicited by nonimmune marrow cells differed sharply from each other under the conditions of limiting dilution assays. The frequencies of anti-chicken responses in recipients of different numbers of marrow cells conformed to the predictions of the Poisson model, while the frequencies of anti-sheep responses did not. Hence, the function of certain marrow-derived cells was expressed differentially during the two immune responses, to exclude that the same precursor units generated anti-sheep or anti-chicken PFC. The former precursor cells or units were functionally more heterogeneous than the latter. Immunization of marrow donors against sheep erythrocytes did not alter the population of cells engaged in anti-chicken responses, since limiting dilution assays with immune and nonimmune marrow cells gave identical results.

Actual Tags :

Sideroblastic, Congenital Nonspherocytic, Immobilized, **Cultured**, Chromium Isotopes, Abnormal, European Continental Ancestry Group, Glucose Oxidase, **Glucosephosphate Dehydrogenase**, Glucosephosphate Dehydrogenase Deficiency, Glutathione, Heinz Bodies, Hematocrit, **Abnormal**, **Humans,** Hydrogen Peroxide, Infection, Mononuclear, Methemoglobin, **Methods**, Oxygen Consumption, Phagocytosis, **Time Factors**, Transaminases

Predicted Tags :

Glucosephosphate Dehydrogenase, Thin Layer, **Humans**, Zucker, Carbon Isotopes, Female, Male, Video, Transmission, Transgenic, Hydrogen-Ion Concentration, Serum-Free, Rabbits, Glucose, **Abnormal**, Starch Gel, Research, Escherichia coli, Missense, Sulfur, Kinetics, **Culture Techniques**, Tritium, Physical, **Time Factors**, **Methods**

CHAPTER V

CONCLUSION AND DISCUSSION

We have proposed an effective solution for MeSH semantic indexing, for rich semantic information in biomedical documents. We developed DeepMeSH, which effectively utilizes dense semantic representation. DeepMeSH must however, in future integrate a variety of diverse evidence. The performance of DeepMeSH can be attributed to the factor: (i) the deep semantic representation, D2V, that has the power of dense representation D2V. Our results also show that LSTM performs better than SimpleRNN and other gated RNNs (GRU). Increasing the amount of data for training improves performance which satisfies the fact that deep neural network needs a large amount of data for it’s training because of the huge number of parameters it has.

The problem of overfitting also arises while training and so dropout (regularizer) was used for regularization. The limitation of this model is the amount of time needed for its training. Large-scale biomedical indexing might need sufficient resources for better performances.

Our experiments indicate that D2V is a very useful representation for finding semantically similar citations. Finding similar citations is a core task in biomedical text mining for knowledge discovery, such as document searching, document clustering and query expansion. Currently, the most widely used method for finding similar citations practically in life science is PRA by NLM, which is based on sparse representation. Thus our new representation will be useful for many applications including searching similar citations and may find more promising applications as well.

CHAPTER VI

REFERENCES

Shengwen Peng, Ronghui You, Hongning Wang, Chengxiang Zhai, Hiroshi Mamitsuka and Shanfeng Zhu. DeepMeSH: deep semantic representation for improving large-scale MeSH indexing.

Yan Yan 1, Xu-Cheng Yin 2, Bo-Wen Zhang 2, Chun Yang 2 and Hong-Wei Hao. Semantic indexing with deep learning: a case study

Nelson,S.J. et al. (2004) The MeSH translation maintenance system: structure, interface design, and implementation. Medinfo, 11, 67–69.

Stokes,N. et al. (2010) Exploring criteria for successful query expansion in the genomic domain. Inf. Retrieval, 12, 17–50.

Lu,Z. et al. (2010) Evaluation of query expansion using MeSH in PubMed. Inf. Retrieval, 12, 69–80.

Gu,J. et al. (2013) Efficient semi-supervised MEDLINE document clustering with MeSH semantic and global content constraints. IEEE Trans. Cybern., 43, 1265–1276.

Huang,M. et al. (2011a) Recommending mesh terms for annotating biomedical articles. J. Am. Med. Inf. Assoc., 18, 660–667.

Huang,X. et al. (2011b) Enhanced clustering of biomedical documents using ensemble non-negative matrix factorization. Inf. Sci., 181, 2293–2302.

Peng,S. et al. (2015) The fudan participation in the 2015 bioasq challenge: Large-scale biomedical semantic indexing and question answering. In: CLEF (Working Notes).

Rios,A. and Kavuluru,R. (2015) Convolutional neural networks for biomedical text classification: application in indexing biomedical articles. In: BCB, pp. 258–267.

Huang,M. et al. (2011a) Recommending mesh terms for annotating biomedical articles. J. Am. Med. Inf. Assoc., 18, 660–667.

Huang,X. et al. (2011b) Enhanced clustering of biomedical documents using ensemble non-negative matrix factorization. Inf. Sci., 181, 2293–2302.

Jiang,J. and Zhai,C. (2007) An empirical study of tokenization strategies for biomedical information retrieval. Inf. Retrieval, 10, 341–363.

Jimeno-Yepes,A. et al. (2012a). MEDLINE MeSH indexing: lessons learned from machine learning and future directions. In: Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium. ACM, pp. 737–742.