Capturing Researcher Expertise through MeSH Classification

Yong-Bin Kang
Faculty of Information Technology
Monash University
Australia
yongbin.kang@monash.edu

Yuan-Fang Li
Faculty of Information Technology
Monash University
Australia
yuanfang.li@monash.edu

Ross L. Coppel
Department of Microbiology
Monash University
Australia
ross.coppel@monash.edu

ABSTRACT

For a large research institution and a broad research discipline such as the life sciences, it is a highly important and very challenging task to *capture* each researcher's expertise, and to *match* researchers by expertise to assist in identifying inter-disciplinary collaboration opportunities and in making informed policy decisions. The challenges are multi-dimensional, stemming from the needs to (a) provide thorough coverage of the breadth and depth of the disciplinary areas, (b) develop accurate representation of researcher's expertise, and (c) process large volumes of data efficiently. Medical Subject Headings (MeSH), a comprehensive taxonomy for the life sciences, has been widely used for indexing MEDLINE publications. In this paper, we present a novel framework for capturing and matching research expertise based on knowledge encoded in MeSH. Specifically, (1) we design a novel and effective hybrid MeSH classification algorithm by combining state-of-the-art methods, and (2) using MeSH terms aggregated from a researcher's publications, we design a researcher matching algorithm based on *semantic similarity* that takes into consideration the structure of the MeSH taxonomy.

Categories and Subject Descriptors

H.3.1 [**Information Systems Applications**]: Content Analysis and Indexing; I.2.6 [**Artificial Intelligence**]: Learning—*Knowledge acquisition*

General Terms

Algorithms, Design

Keywords

MeSH, Researcher profile, Expertise matching

1. INTRODUCTION

In a large research institution, there may be thousands of researchers working in a diverse array of disciplines, working on different, but relevant sub-disciplinary areas. A sound understanding of researchers' expertise and a mechanism for systematically representing such expertise can have a significant impact on a number of issues, such as identifying (individual or institutional) collaboration partners and decision making.

Inter-disciplinary collaboration can and does arise organically. Being able to identify research expertise and identifying complementary research expertise further facilitate collaboration within and across disciplinary and institutional boundaries.

Moreover, the identification of individual researcher's expertise is the foundation of understanding and evaluating the research strengths of an organisational unit, such as a lab, a department, a faculty or an institution. This knowledge in turn better informs research planning and strategic decision making.

A number of challenges need to be tackled to achieve the above tasks. Firstly, research topics and subjects need to be described *comprehensively* in the interested branches of study. Secondly, researchers' expertise needs to be represented *accurately*. Thirdly, large volumes of data, primarily in the form of publications, needs to be processed *efficiently*.

These challenges manifest in the life sciences, which encompass studies in a very large number of branches of science including biology, medicine, as well as inter-disciplinary areas such as bioinformatics. MEDLINE, one of the largest databases for the life sciences, now provides references to more than 21 million biomedical articles. The number of articles in MEDLINE continues to rapidly increase, at about 700k articles per year. Therefore, the breadth and depth of the fields, as well as the large volumes of publications data make the life sciences an especially challenging domain for representing researcher expertise.

The Medical Subject Headings (MeSH) taxonomy provides a comprehensive common vocabulary for describing research topics relevant to the life sciences, containing 26k+ terms (the MeSH main headings) in its current iteration. The MeSH vocabulary has been extensively used to classify (index or annotate) biomedical articles in MEDLINE. We believe its broad coverage and wide adoption make MeSH an ideal candidate as a basis for the representation of researcher expertise.

However, the task of classifying articles using MeSH terms, often referred to as *MeSH classification*, is a very challenging problem, given the large sizes of MEDLINE and MeSH and the complex hierarchical structure of MeSH. As a result, despite recent development in automatic methods [3, 2, 12], MeSH classification is still performed semi-automatically. This process requires extensive domain knowledge, is expensive, time-consuming and error-prone.

K-CAP 2015, October 07-10, 2015, Palisades, NY, USA.
 ISBN 978-1-4503-3849-3/15/10 $15.00.
DOI: http://dx.doi.org/10.1145/2815833.2815837 .

In this paper, we tackle the important task of representing researcher expertise in the life sciences domain, and develop algorithms for all the main underpinning research questions: automated MeSH classification. Based on this algorithm we also develop methods for matching researchers based on their expertise. Our main contributions are two-fold:

- HyClass, a novel hybrid MeSH classification framework that combines three widely-used methods. Our evaluation using a large benchmark dataset shows that, on the task of classifying using the entire MeSH taxonomy, HyClass is effective and outperforms a widely-used, state-of-the-art method, Medical Text Indexer (MTI) [3]. Compared to LRankCF, another state-of-the-art system capable of performing classification using the entire MeSH taxonomy, HyClass also shows higher improvement ratio over MTI.
- A researcher matching algorithm with respective to their expertise, based on the aggregation of MeSH classification of researchers' published articles and exploiting the structure of MeSH.

Moreover, our proposed framework of representing and matching researcher expertise has broad application beyond just the life sciences: any discipline with a shared, comprehensive taxonomy describing disciplinary topics can utilise this framework. These disciplines and respective taxonomies include Computer Science with the ACM Computing Classification System (CCS, `http://www.acm.org/about/class/`), Engineering with the IEEE Thesaurus (`http://www.ieee.org/publications_standards/publications/services/thesaurus2.html`), Mathematics with Mathematics Subject Classification (MSC, `http://msc2010.org/`), and Physics with the Physics and Astronomy Classification Scheme (PACS, `http://www.aip.org/publishing/pacs`), to name a few.

2. BACKGROUND AND RELATED WORK

Scientific endeavours are increasingly conducted in an interdisciplinary manner [19]. A large body of research, in scientometrics, has developed methods of analysing and mapping scientific research [17], especially on a global scale [7]. However, from an institutional perspective, there is also an urgent need for a better understanding of research expertise of individual researchers, organisational units as well as the institution itself as it can facilitate research collaboration and enable informed decision making in research strategies and policies.

A large number of tools and systems have been proposed to measure, analyse, map and visualise research specialty and expertise [7], usually based on bibliographical information and collaboration and/or citation networks.

2.1 MeSH Classification

Classification of articles using controlled vocabularies is a common practice in a number of disciplines, where an article is assigned a number of terms from the vocabulary to denote its major research topics. In the life sciences, MeSH is a large taxonomy developed and maintained by the United States National Library of Medicine (NLM) as a shared vocabulary for the classification of research articles. Its widespread adoption and comprehensive coverage of the domain enable MeSH classification to be employed to represent researchers' expertise.

However, human experts still need to review automatically generated MeSH terms for MEDLINE articles. It requires extensive domain knowledge from subject experts, and is time-consuming and error-prone. To assist and accelerate the semi-automatic MeSH classification process, NLM first introduced an automatic indexing tool, MTI [3], which has lead to the development of various methods for improving automatic MeSH classification.

Most recent approaches to MeSH classification can be divided into two broad categories: *text mining scheme*, and the *machine learning scheme*.

The text mining scheme aims to identify particularly useful features from training articles and use them for MeSH classification. The premise is that such features can well characterize the contents of the articles and help to identify strongly evident and interesting relationships between the features and the articles. Vasuki and Cohen [27] employed a term-document matrix using Reflective Random Indexing (RRI) to represent such relationships, where rows represent articles and columns represent terms derived from training articles. Wahle et al. [28] proposed a refinement of RRI that uses an entry with a binary-valued weight instead of a real-valued weight to improve performance for finding the neighbors. Jimeno-Yepes et al. [13] shows that a combination of MEDLINE citations (using only title and abstract) and different types of summaries generated automatically from full text articles could improve the recommendations suggested by MTI. Kavuluru and He [15] proposed to extract concepts from the UMLS Metathesaurus from a target article, convert the concepts to MeSH term candidates in UMLS, and finally recommend those candidates whose co-occurrence frequencies are above a threshold for MeSH classification. Such frequencies are found from a large corpus of articles (22 million articles) previously indexed by domain experts.

The machine learning scheme focuses on learning from the training articles indexed with ground truth MeSH terms. The learned models are used to assign those MeSH terms with higher relevance scores to target articles. Citation information is exploited [2] to enrich the feature set. A binary classifier (e.g. Naïve Bayes) is then trained to learn which MeSH terms are associated with particular training articles based on the proposed feature representation. This system is evaluated on the task of classifying articles using only the top-20 most frequent MeSH terms in a benchmark dataset, and shows good performance. Huang et al. [12] proposed to use a learning-to-rank approach (hereafter referred to as LRankCF), where it learns how to assign a relevant score to a MeSH term from training articles. Given a target article, it finds its neighbouring articles, extracts a list of MeSH terms from these articles, and finally assigns a relevance score to each of these terms for the target article. Evaluated for the task of classification over the entire MeSH taxonomy on a large benchmark dataset, LRankCF shows superior performance over existing systems and it represents the state-of-the-art.

Classification over the entire MeSH taxonomy is a more useful task than over the top-20 most frequent MeSH terms as it allows a more nuanced characterisation of a publication's topics. This is especially true as half of the top-20 most frequent terms from [2] (as shown in Figure 1) are very generic terms (such as Mutation, Cells cultured and Calcium), residing on the 2nd or 3rd level in the MeSH taxonomy. On the other hand, it is also more challenging as a classification method needs to handle a much larger number of class labels (26k vs 20). In this paper, we tackle the problem of classification over the entire MeSH taxonomy, and we will compare against LRankCF in our evaluation.

2.2 Researcher Matching

Most previous approaches to researcher (expert) matching focused on measuring matching scores between a researcher and a query for applications such as paper-review assignment and

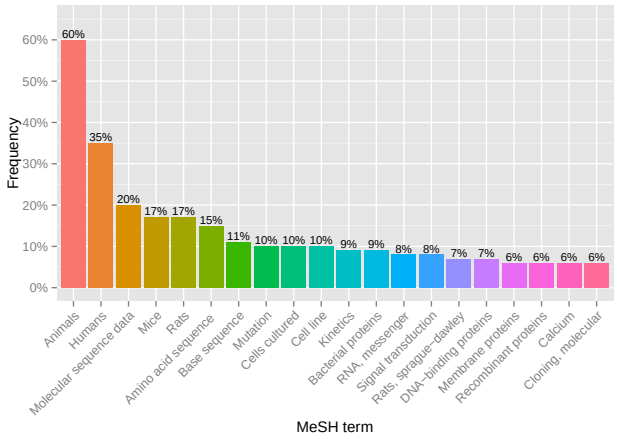

Figure 1: The top-20 most frequent MeSH terms in [2].

product-review assignment [25]. An expert is often represented using a set of its publications and a query is often represented in the form of topics extracted from a document using topics models [29] or language models [9]. The matching scores were mostly measured by probabilistic models such as keyword matching, latent semantic indexing or probabilistic topic modeling [25].

Our work differs from previous approaches in some key aspects. (1) Our algorithm is focused on researcher-researcher matching, but not expertise-expertise matching. (2) our algorithm is based on semantic similarity between terms in a taxonomy, and it incorporates semantic knowledge of the taxonomy. On the contrary, previous approaches mainly rely on free text, hence they do not make use insights into the target domain provided by the taxonomy. (3) Our matching algorithm also considers, formalises and utilises the weights of researcher expertise to be compared. (4) Our algorithm can be easily adapted to other domains where domain knowledge exists in the form of taxonomies.

3. HyClass: A HYBRID MESH CLASSIFICATION FRAMEWORK

In this section, we propose HyClass, a hybrid MeSH classification method. Given a target article x, HyClass is able to predict relevance scores of MeSH terms with respect to x, and rank these MeSH terms by their relevance scores. The relevance and ranking form the foundation of MeSH classification.

Our hybrid method makes use of *recommender systems*, which recommends items (i.e., MeSH terms in this context) to users. Recommender systems can generally be divided into two categories: *collaborative* methods (CF) based on user similarity and *content-based* methods (CB) based on item similarity. Specifically, we combine MTI, the baseline MeSH classifier developed by the NLM, with a collaborative recommender, denoted R_{cf} and a content-based recommender, denoted R_{cb}. R_{cf} recommends MeSH terms for x based on MeSH terms used to classify similar articles to x. R_{cb} recommends MeSH terms for x based on a model learned from the training articles using a multi-label classifier.

Formally, let $\mathcal{M}$ be the set of MeSH terms for classification. Let $r(x,m)$ be a function that measures the relevance score of a MeSH term $m \in \mathcal{M}$ for a target article x; $r(x,m)$ is normalised into $[0,1]$ (from completely irrelevant to completely relevant). The goal of of our method is to return a ranked list of MeSH terms $\{m_1,m_2,...,m_k\} \subseteq \mathcal{M}$ to the article x, where $r(x,m_i) \geq r(x,m_j)$ for all $i<j$.

3.1 Data Representation

The objective of both R_{cf} and R_{cb} is to predict relevance scores of MeSH terms for a target article using *training data* consisting of biomedical articles with human-indexed MeSH terms. We represent the training data as a matrix, where rows represent articles and columns consist of two spaces:

- *feature space* holds the set of the most *informative* features, where each entry indicates how important a feature is with respect to an article and takes a real-value in [0,1], and
- *class space* holds the set of MeSH terms used to classify each article, where each entry takes a binary value indicating whether a MeSH term has been used to classify an article (denoted by 1) or not (denoted by 0).

To formally define the representation of the training data, the following notations will be used in the rest of this paper:

- Let $\mathcal{M}$ be the set of terms from the MeSH taxonomy.
- Let $\mathcal{A}$ be the set of training articles.
- Let $\mathcal{F}$ be the set of all possible features that can be extracted from $\mathcal{A}$.
- Let $F \subseteq \mathcal{F}$ be the finite set of the most informative features that carry primary information about articles in $\mathcal{A}$.

Given the above definitions, the training data $\mathbf{D}$ can be formally represented as an $|\mathcal{A}| \times (|F|+|\mathcal{M}|)$ matrix. Each row r_i represents a combination of the feature space and class space for article $a_i \in \mathcal{A}$:

$$r_i = (\underbrace{(w_{i,1},...,w_{i,|F|})}_{\text{feature space}},\underbrace{(c_{i,1},...,c_{i,|\mathcal{M}|})}_{\text{class space}}) \quad (1)$$

where $w_{i,j} \in [0,1]$ indicates the informativeness of a feature f_j for an article a_i, and $c_{i,k} \in \{0,1\}$ indicates whether a MeSH term m_k is used (denoted by 1) or not (denoted by 0) to classify a_i.

For features $\mathcal{F}$, HyClass can use any types of terms or concepts derived from the training articles $\mathcal{A}$ that can effectively characterize $\mathcal{A}$. The Unified Medical Language System (UMLS) [16] is a large collection of comprehensive controlled vocabularies, including MeSH in the life sciences that also provides mappings across the vocabularies. In this work, we use UMLS concepts as features recommended by SemRep [22] from the title and abstract of the articles. The premise is that these features can be suitable for capturing essential semantic representation of $\mathcal{A}$ and thus effective in representing the topics of $\mathcal{A}$ precisely. To determine the informativeness of a feature $f_j \in \mathcal{F}$ for each article $a_i \in \mathcal{A}$, we apply the widely used TF-IDF measure [23]. According to this measure, the k most informative features are selected and used to represent the training data $\mathbf{D}$.

3.2 Collaborative Method (R_{cf})

R_{cf} takes three main steps. In the first step, given a target article x, its features are extracted. These features are then represented as a vector, denoted as $\vec{x}$, using the most informative features $F \in \mathcal{F}$. For x, the informativeness of a feature $f_j \in F$ is calculated using TF-IDF from the training data $\mathbf{D}$, where $t_{i,j}$ denotes the number of times that f_j appears in x.

In the second step, a set of nearest neighbouring articles of x is identified in $\mathbf{D}$. To find them, we adopt the widely used similarity measure, *Pearson correlation coefficient* [23]. More specifically, given x, we use its vector $\vec{x}$ to find its similar articles in $\mathcal{A}$. Each article $a \in \mathcal{A}$ to be compared is also represented as a vector $\vec{a}$ using only its feature space. Then, the similar-

ity $sim(x,a)$ between two articles x and a is calculated as the following similarity $sim(\vec{x},\vec{a})$ between two vectors $\vec{x}$ and $\vec{a}$:

$$sim(\vec{x},\vec{a}) = \frac{\sum_j (e_j - \overline{e})(e'_j - \overline{e'})}{\sqrt{\sum_j (e_j - \overline{e})^2 \sum_j (e'_j - \overline{e'})^2}} \quad (2)$$

where e_j (resp. e'_j) is the j-th value in $\vec{x}$ (resp. $\vec{a}$), $\overline{e} = \sum_{j=1}^{|\vec{x}|} e_j$ (resp. $\overline{e'} = \sum_{j=1}^{|\vec{a}|} e'_j$) is the average of $\vec{x}$ (resp. $\vec{a}$). This similarity indicates the ratio of the angle of the relationship between $\vec{x}$ and $\vec{a}$ to the standard deviations of these two vectors. It is normalized into [-1,1]. The closer a value is to 1, the higher the similarity is. Using this equation, we say that two feature vectors are similar if their similarity is above a threshold 0.5, often deemed as the minimum value of a strong correlation [8].

In the final step, once we find the most similar articles $\mathrm{NH}(x)$ for a target article x, R_{cf} recommends a ranked list of MeSH terms $\mathrm{R}_{cf}(x)$ with their relevance scores to classify x using $\mathrm{NH}(x)$. Formally, the relevance score $r_{cf}(x,m)$ for a MeSH term m with respect to x in R_{cf} is defined as

$$r_{cf}(x,m) = \frac{\sum_{e \in \mathrm{NH}(x)} sim(\vec{x},\vec{e}) * r'(e,m)}{\sum_{e \in \mathrm{NH}(x)} sim(\vec{x},\vec{e})} \quad (3)$$

where $sim(\vec{x},\vec{e})$ is the similarity between x and $e \in \mathrm{NH}(x)$; and $r'(e,m)$ is the relevance score of m with respect to e—$r'(e,m)$ is 1 if m is used to classify e, and 0 otherwise. Note that the more similar x and e are, the more $r'(e,m)$ can contribute to the calculation $r_{cf}(x,m)$.

3.3 Content-Based Method ($\mathbf{R}_{cb}$)

In R_{cb}, the relevance score of a MeSH term for a target article is estimated by a *model* learned from the training data $\mathbf{D}$ using a multi-label (ML) classifier. This classifier learns how known particular MeSH terms (i.e. multi-labels) are associated with the most informative features $F \in \mathcal{F}$ of articles in $\mathbf{D}$ and then is used to predict highly relevant MeSH terms for target articles.

ML classification methods can be divided into two approaches [26]: (1) problem transformation methods that transform the classification task into one or more single-label (SL) classification tasks, and (2) algorithm adaptation methods that extend specific SL classification algorithms to directly accommodate ML data. In this work, RAkEL, a representative of the first approach, is chosen as the ML classifier for R_{cb}, due to its highly predictive performance and computational efficiency with a large number of of labels [26].

RAkEL randomly divides a set of labels (e.g. MeSH terms) into a set of small k-labelsets, and trains a SL classifier on each k-labelset using LP. Thus, RAkEL learns t SL classifiers $h_1,...,h_t$ using t disjoint labelsets $L_{j[1,t]}$ ($\bigcap_{j=1}^{t} L_j = \emptyset$) using LP. LP is a transformation method that considers each k-labelset as a distinctive class in a new SL classification task. To classify a target instance, $h_{j[1,t]}$ thus finds the most probable class (i.e. a set of labels) from L_j. Eventually, RAkEL performs t SL classifiers $h_1,...,h_t$ to make predictions for different k-labelsets, and then combines their predictions. As a SL classifier, we choose RandomForest (RF) [5], which is an ensemble of randomized decision trees, due to its its ability to run efficiently on large datasets [11].

In our context, given the training data $\mathbf{D}$, RAkEL learns t RF classifiers $h_1,...,h_t$ using LP from different k-labelsets randomly selected from $M \in \mathcal{M}$, the set of MeSH terms used to classify the training articles $\mathcal{A}$. Each RF classifier $h_{j[1,t]}$ learns:

$$h_j : \mathcal{A} \rightarrow 2^M \quad (4)$$

where each instance of $\mathcal{A}$ is represented a feature vector using TF-IDF, and 2^M is the powerset of M. In our study, k for k-labelset is chosen as 3 and t is chosen as $|M|$ as optimal values, according to the suggestion in [26].

Given a new target article x, the binary predictions of $h_i(x,l_j)$ of all SL classifiers h_i for all labels $l_j \in L_i$ are aggregated to make the final predictions about the degree of relevance of all labels in L^t denoted as $\bigcup_{i=1}^{t} L_i$. The mean of each prediction for each label $l_j \in L^t$ is calculated and used as the relevance score of l_j for x. Formally, the relevance score $r_{cb}(x,l_j)$ for a label (i.e. a MeSH term) $l_j \in L^t$ with respect to x in R_{cb} is computed as

$$r_{cb}(x,l_j) = \frac{1}{t} \sum_{i=1}^{t} h_i(x,l_j) \quad (5)$$

where t is the number of the SL classifiers, and $h_i(x,l_j)$ is calculated if l_j is seen in the corresponding k-labelset L_i. Using the relevance scores of all labels in L^t, R_{cb} finally recommends a ranked list of MeSH terms $\mathrm{R}_{cb}(x)$ with their relevance scores to classify x.

3.4 Our Hybrid Classification Framework (HyClass)

A hybrid recommender often combines a CF and a CB in some ways to gain complementary benefits, eventually leading to better recommendation performance [1]. Motivated by this, we independently and linearly combine the MeSH terms with their relevance scores that have been recommended from both R_{cf} and R_{cb}. That is, the capabilities of both recommenders are independently exploited to make final MeSH term predictions in a straightforward way (i.e. linearly). This approach has been shown effective in building a composite hybrid recommender with multiple recommenders [4]. Another feature of our hybrid recommender, HyClass, is the incorporation of MTI. Our objective is to independently combine individual recommendation scores of MTI, in addition to R_{cf} and R_{cb}, to improve the quality of the overall MeSH term prediction.

Formally, a general definition of HyClass can be described as follows. Let $RC = \{rc_1, rc_2, ..., rc_n\}$ be the set of the all recommenders combined in HyClass. Thus, in this work, RC contains R_{cf}, R_{cb} and MTI respectively, and $|RC| = 3$. Also, let $rc_{i[1,n]}(x)$ be a ranked list of MeSH terms recommended by rc_i for a given target article x. Then, given x, HyClass generates a ranked list of MeSH terms, $\mathrm{HyClass}(x) = \{m_1, m_2, ..., m_z\} \in \mathcal{M}$, according to their final relevance scores to x (i.e. $r(x,m_i) \geq r(x,m_j)$ for all $i < j$). The relevance score of a MeSH term $m_{i[1,z]} \in \mathrm{HyClass}(x)$ with respect to x is computed as:

$$r(x,m_i) = \frac{\sum_{j=1}^{|RC|} w_j * rc_j(x,m_i)}{\sum_{j=1}^{|RC|} w_j}, \ \forall m_i \in \bigcup_{j=1}^{|RC|} rc_j(x) \quad (6)$$

where $rc_j(x,m_i)$ is the relevance score for m_i with respect to x determined by recommender rc_j. The notation w_j represents a weight of the corresponding recommender rc_j, indicating how much $rc_j(x,m_i)$ contributes to the calculation of the relevance score $r(x,m_i)$. Note that the relevance scores of MeSH terms recommended by rc_1 (R_{cf}) and rc_2 (R_{cb}) are normalized into a real value in [0,1]. On the other hand, raw MTI scores directly recommended from MTI (rc_3) are not normalized. To normalize the scores, we first transform them into their log-scale scores

and then divide them by the respective maximum from a target article. Here, the log-transformation is applied to mitigate the difference between each score and the maximum.

The definition of an optimal value of each $w_{j[1,n]}$ requires complete knowledge of the real recommendation capabilities of each recommender rc_j, which is not available. Thus, in this work, we assume that all recommenders in RC are equally important to each other, i.e., $w_1 = w_2 = ... = w_{|RC|}$, where $\sum_{j=1}^{|RC|} w_j = 1$. This assumption is also consistent with the general assumption in a linearly weighted hybrid recommendation approach [6].

4. MESH CLASSIFICATION EVALUATION

We evaluate HyClass on the task of classification of the entire MeSH taxonomy using the dataset previously used to evaluate the state-of-the-art system LRankCF. The evaluation is conducted on a NLM dataset (hereafter referred to as NLM2007 [1]) which has been often used as a benchmark to evaluate MeSH classifiers [12, 27]. Additionally, MTI is compared as a strong baseline method. In our evaluation, HyClass shows superior performance over both MTI and LRankCF.

Our evaluation methodology follows that of LRankCF, i.e., evaluating performance based on the top-25 ranked MeSH terms for each article in NLM2007. We use the same performance metrics as in LRankCF, which are F-measure and Mean Average Precision (MAP). More details on the metrics can be found in [12].

We intend to compare HyClass with LRankCF on the same training and testing sets.[2] To that end we use the same 200 test articles from NLM2007 that is used to evaluate LRankCF. To train HyClass, we employ the same training set for LRankCF, a collection of 10k articles that consist of 50 most similar articles of each article in NLM2007 (the *neighbouring articles* in LRankCF). However, we note that LRankCF uses two additional training sets: (1) 13k+ articles to extract a number of meta-level features required to train a ranking algorithm (i.e., ListNet), and (2) 10k articles to train LRankCF for ranking purposes.

As HyClass is based on MTI, we can only use the 2013 version of MeSH that the current MTI supports, where LRankCF recommends terms from the 2010 version of MeSH, which contains close to 5% fewer main headings (terms). Moreover, LRankCF is evaluated against the 2010 version of MTI (denoted MTI_{2010}), and we can only compare with the current version of MTI (denoted MTI_{2013}). Moreover, surprisingly, as can be seen in Table 1, the performance of the more recent MTI_{2013} is sufficiently worse than that of the older MTI_{2010}.

Due to the above differences, a completely fair comparison between HyClass and LRankCF is not possible. Moreover, given that MTI is a key component in HyClass, the performance of HyClass is directly impacted by the performance of MTI. Therefore, we compare the two systems indirectly, by their *improvement ratios* of F-measure and MAP over their respective baseline MTI.

In the training set, the total number of UMLS features is 28k. We only consider feature sets consisting of the 100–1,000 most informative features with increments of 100 from such a large feature space. In total, 9,696 classes (i.e. MeSH terms) are actually used in the training set. However, we observe that the distribution of the document-frequency (DF) rates of these terms is highly imbalanced. The DF rate of each MeSH term represents the rate of the number of articles that contain the term over the total number of the articles in the training set. For example, the DF rates of 7,555 classes are less than 0.02% (i.e. DF = 2), while the DF rates of only 429 classes are above 0.5% (i.e. DF = 50).

Typically, such an imbalanced distribution of classes causes unsatisfactory classification outcomes for classifiers, since target articles are much likely to be classified into majority classes. To avoid this problem, one way is to ignore minor classes that rarely appear in the training set [20]. Thus, for R_{cb}, we restrict the number of MeSH terms to be recommended to 200, and the DF rate of each of these 200 MeSH terms is more than or equal to 0.8% (i.e., DF = 80 as there are 10,000 articles in the training set).[3]

We now compare the performance of HyClass with the result of LRankCF directly obtained from [12], as well as with MTI_{2013}. Table 1 summarises the results. According to Wilcoxon signed rank test at 99.9% confidence, we find that HyClass significantly outperforms MTI_{2013} in terms of F-measure and MAP as indicated by 'bullets'.

As previously described, we compare them based on the improvement ratio over their respective baseline (MTI_{2010} and MTI_{2013}) in terms of F-measure and MAP. In other words, we compare LRankCF with MTI_{2010} as directly obtained from [12]. On the other hand, we compare HyClass with MTI_{2013}. Due to the difference of the MeSH versions, the results for MTI are slightly different, as can be seen from the table. Finally, as can be observed, HyClass is competitive to LRankCF in terms of F-measure and MAP. In particular, we observe that HyClass significantly outperforms LRankCF in terms of improvement ratio of both F-measure and MAP, noticeably much better in MAP whose improvement ratio is 48% compared to 39% for LRankCF.

Table 1: The comparison between HyClass and LRankCF.

Method	Raw Result		Improvement ratio	
	F-measure	MAP	F-measure	MAP
MTI_{2010}	0.409	0.450	N/A	N/A
LRankCF	0.504	0.626	23%	39%
MTI_{2013}	$0.389^{\bullet}$	$0.373^{\bullet}$	N/A	N/A
HyClass	0.489	0.553	**26%**	**48%**

Through the evaluation on classification over the entire MeSH taxonomy, we observe that HyClass significantly outperforms the current version of the widely used baseline systems MTI. It also outperforms the state-of-the-art LRankCFin terms of improvement ratio.

5. AN RESEARCHER MATCHING ALGORITHM BASED ON SEMANTIC SIMILARITY

Matching researchers by their expertise is an important way to identify research strength and facilitate potential collaboration opportunities. This capability is especially useful for researchers in different faculties, institutions or across disciplinary areas.

A number of important problems need to be addressed in researcher matching. Specifically,

1. How can we represent the expertise of a researcher in a target domain?
2. How can we measure the weight (importance) of each piece of expertise with respect to a research in the target domain?

[1] `http://ii.nlm.nih.gov/Eval_Analysis/Eval_2007/listings.shtml`.

[2] The training and testing sets are available at `http://www.ncbi.nlm.nih.gov/CBBresearch/Lu/indexing/`.

[3] We note that 200 is the optimal value of MeSH terms between [100,1000] with increments of 100.

3. How do we define a matching score between two researchers in the target domain?

A researcher's expertise can be naturally derived from her publications, which we use as the foundation to represent researcher expertise. Moreover, we argue that, for a domain with a comprehensive taxonomy, it is natural and feasible to represent researcher expertise through an aggregation of weighted terms collected from her publications.

In this section, we address the researcher expertise matching task by addressing the above three problems. First, we represent the expertise of a researcher using MeSH terms that *classify* her publications (Section 5.1). Second, we propose a method that measures the weight of each MeSH term (i.e., a piece of expertise) with respect to a researcher using its number of occurrences in her publications (Section 5.2). Third, we propose an algorithm that defines and calculates a matching score between two researchers by aggregating matching scores between MeSH terms for each of the researchers (Section 5.3). To measure a matching score between two MeSH terms, a key idea is to combine a *semantic similarity* between them and their weights with respect to each of the researchers.

It is worth noting that our method is not specific to the life sciences (with MeSH). It is applicable in any domain with a comprehensive taxonomy, such as computer science (with ACM CCS), engineering (with the IEEE Thesaurus) and physics (with PACS), as discussed in Section 1.

5.1 Representing Expertise of a Researcher

As in HyClass, MTI is used as input for our researcher matching algorithm to generate MeSH terms from her list of publications. Essentially, the expertise of a researcher is represented by the *multiset* of MeSH terms aggregated from her publications. This is due to the fact that given a publication, it is possible for a MeSH term to appear more than once as MeSH terms are extracted by MTI from each sentence of the contents of a publication.

Let E be the set of researchers (experts). Let P be the set of all publications published by experts in E, and P_e be all publications published by expert $e \in E$. Let M be the set of all possible MeSH terms that annotate (classify) publications in P. Then, we represent the expertise of an expert e as a set of MeSH terms that annotate the publications P_e.

Given a publication $p \in P$, we denote by $t_p : M \to \mathbb{N}$ the *multiset* of MeSH terms used to annotate p, expressed as a function from the set of MeSH terms M to natural numbers $\mathbb{N}$.

For example, given a publication p, assume that it has three sentences in the abstract. The MeSH terms that annotate p can be represented by the following set of sets of MeSH terms, one for each sentence: $\{\{\mathtt{m}_1,\mathtt{m}_2,\mathtt{m}_3\},\{\mathtt{m}_1,\mathtt{m}_2\},\{\mathtt{m}_2,\mathtt{m}_3\}\}$.

Hence, the multiset t_p can be represented by $t_p = \{\{\mathtt{m}_1,\mathtt{m}_2, \mathtt{m}_3,\mathtt{m}_1,\mathtt{m}_2,\mathtt{m}_2,\mathtt{m}_3\}\}$. Or alternatively, $t_p = \{(\mathtt{m}_1,2),(\mathtt{m}_2,3),(\mathtt{m}_3,2)\}$

5.2 Measuring the Weight of Expertise

As each publication is usually annotated by a number of MeSH terms, and each researcher usually has expertise in a number of topics represented by MeSH terms, it is important to measure the relative importance of these terms for a given researcher.

Our key idea for measuring the *weight* (importance) of each piece of expertise (i.e., each MeSH term) of an expert $e \in E$ is based on its number of occurrences across the publications of e, i.e. P_e. Our objective is to generate a ranked list of MeSH terms for e. For this purpose, we define a function $f_{m,e}$ that measures the weight of a MeSH term in M, used to annotate one or more publications in P_e, for e, i.e.

$$f_{m,e} : M \times E \to [0,1] \tag{7}$$

where the weight is a real number normalised between [0,1]. The higher the weight of MeSH term m is, the more relevant it is to researcher e.

The weight of m for e, $f_{m,e}$, is obtained through aggregation from P_e, the set of all publications of e. We define a function $f_{m,p,e}$ that measures the weight of a MeSH term $m \in M$ within a publication $p \in P_e$ for an expert $e \in E$ as follows:

$$f_{m,p,e} = \frac{t_p(m)}{\max_z t_p(z)} \tag{8}$$

where $t_p(m)$ is the number of times that the MeSH term m appears in the publication p as defined above, and the maximum is computed over $t_p(z)$ over all MeSH terms $z \in M$ that appear in p. Recall that as described in Section 5.1, each MeSH term can appear more than once in a given publication.

Once we define the function $f_{m,p,e}$, the function $f_{m,e}$ in Eq. 7 is defined more formally as follows taking into account all publications of e, i.e., P_e:

$$f_{m,e} = \frac{\sum_{p \in P_e} f_{m,p,e}}{|P_e|} \tag{9}$$

$f_{m,e}$ calculates the weight of MeSH term m for researcher e from each publication $p \in P_e$, averaged across all publications P_e.

5.3 Measuring a Matching Score between Two Researchers

Once we generate a ranked list of MeSH terms of each researcher in the domain, we can make use of it to define a *matching* (*relevance*) score between two researchers. For this, we need to define a function $f_{e,e'}$ given two researchers e and e':

$$f_{e,e'} : E \times E \to [0,1] \tag{10}$$

The higher a weight is, the more the two researchers are matched (relevant), meaning that the more they share common expertise.

In this section, we present a matching algorithm that calculates $f_{e,e'}$. Given two researchers, the basic idea underpinning the algorithm is to *aggregate* matching scores between all pairs of MeSH terms from the two researchers. The key idea in measuring a matching score between two MeSH terms is to combine the following two factors: (1) a *semantic similarity* between the MeSH terms, and (2) the *weights*, calculated in the previous section, of the MeSH terms.

5.3.1 Semantic Similarity

In our context, the definition of the semantic similarity is a function $sim(m_1,m_2) \to [0,1]$, where $m_1 \in M$ and $m_2 \in M$ are MeSH terms to be compared. The range of this function is normalized to real numbers between 0 (completely dissimilar) and 1 (identical). Thus, the similarity should tend to be 1 as m_1 and m_2 have more and more common "characteristics", whose notion is subjective [10]. We define an expertise similarity measure based on domain knowledge captured through the MeSH terms.

The traditional syntactic similarity measures have a common limitation that relies on the notion of *syntactical* differences between the elements to be compared. In other words, these approaches have a limited capability for obtaining reasonable similarity scores due to the disregard of domain knowledge [10].

To improve those traditional similarity approaches, the importance of exploiting available background knowledge about the application domain has been a recent research focus [14, 18].

Intuitively, in a taxonomy such as MeSH, the lower concepts (i.e., MeSH terms) inherit all the characteristics from their superordinate concepts. Namely, the higher the position of a concept in the taxonomy, the more abstract the concept is. In addition, highly related concepts are grouped together and the path between two different concepts in the hierarchy reflects how they are semantically related in the application domain. As a consequence, using semantic knowledge about the taxonomy can provide useful insights of the underlying domain knowledge, thereby facilitating comparative analysis of the concepts for similarity measurement.

Thus, in our work, we leverage the MeSH taxonomy where all of the 26k+ MeSH terms are arranged hierarchically from most general to most specific in up to 12 hierarchical levels. In MeSH, on the first level, MeSH terms are organized into 16 broad categories, including category A "Anatomy", category B "Organisms", and category C "Diseases". Each category is further divided into successive subcategories. Each MeSH term appears in at least one place in the tree, and may appear in additional places where appropriate.

In general, there are two approaches for computing a semantic similarity between two MeSH terms [18]: (1) *distance-based approaches* which estimate the distance between two elements; and (2) *node-based (information content) approaches* which estimate the amount of shared information content between two elements.

However, a main problem with distance-based approaches is that they assume that edges connecting elements in the tree represent uniform distances, i.e., all the semantic edges have the same weight. For example, assuming that in Fig. 2, every edge has uniform distance of 1, the distance between 'Diseases' and 'Organisms' is 2. This distance is identical to the distance between 'Virus Diseases' and 'Neoplasms'. However, intuitively one would judge that the similarity between the latter pair of terms should be higher than the former pair of terms, since the latter pair are siblings, and they are positioned in the lower level in the tree, thus they share more specific information than the former terms.

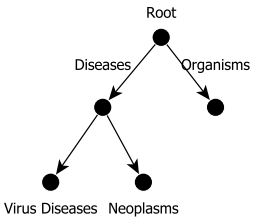

Figure 2: A simple snippet of the MeSH tree

To overcome this problem, we choose a node-based approach. Given a tree, a node represents a unique concept containing a certain amount of information, and an edge represents a relation between two concepts. The similarity between two concepts is then the extent to which they *share* information in common. Namely, the more information they share, the more similar the are. To measure the information content of each MeSH term m using the MeSH taxonomy $\mathcal{M}$, represented as a tree, denoted as $ic(m)$, we use the following method [24]:

$$ic(m) = 1 - \frac{\log(|sc(m)|+1}{\log(|\mathcal{M}|)} \quad (11)$$

where $sc(m)$ is the set of transitively subsumed (descendent) terms of m and $|\mathcal{M}|$ is the size (total number of terms) in $\mathcal{M}$. The denominator is equivalent to the value of the most informative concept, serves as a normalizing factor assuring that $ic(m)$ is in $[0,1]$. A merit of this method is to exploit only structural information of the tree, not relying on corpora analysis as [21], hence more efficient.

Using Eq. 11, we apply Jiang's similarity measure [18] to define the similarity between two MeSH terms. The reason is that several studies commonly found that Jiang's measure is the most effective and outperforms other approaches [24, 18]. This measure is defined using the information content of the compared MeSH terms m_1 and m_2, measured by Eq. 11, as follows:

$$sim(m_1,m_2) = 1 - \frac{ic(m_1)+ic(m_2)-2\cdot ic(lcs(m_1,m_2))}{2} \quad (12)$$

where $lcs(m_1,m_2)$ is the "least common subsumer" (superclass) that subsumes both m_1 and m_2. This formulation normalises the similarity score in interval [0,1].

5.3.2 Researcher Matching Algorithm

We now present our researcher matching algorithm. Let $e_1 = \{m_{1,1},...,m_{1,h}\}$ and $e_2 = \{m_{2,1},...,m_{2,g}\}$ be two researchers (represented by their respective MeSH terms) to be compared. Then, the matching score between e_1 and e_2, $f_{e,e'}(e_1,e_2)$, is computed from both the similarity scores measured by Eq. 11 and the weights of the MeSH terms compared computed by Eq. 7, as follows:

$$f_{e,e'}(e_1,e_2) = \frac{1}{|e_1|\cdot|e_2|} \sum_{\substack{m_{1,i} \\ m_{2,j}}} sim(m_{1,i},m_{2,j}) \cdot \theta(m_{1,i},m_{2,j}) \quad (13)$$

where $m_{1,i} \in e_1, m_{2,j} \in e_2$ and $\theta(m_{1,i},m_{2,j})$ denotes the *strength* of the weights of two terms $m_{1,i}, m_{2,j}$, defined as follows:

$$\theta(m_{1,i},m_{2,j}) = 1 - \frac{|f_{m_{1,i},e_1} - f_{m_{2,j},e_2}|}{2} \quad (14)$$

where $f_{m_{1,i},e_1}$ and $f_{m_{2,j},e_2}$ are computed using Eq. 7. Note that $|f_{m_{1,i},e_1} - f_{m_{2,j},e_2}|$ represents the *absolute value* of $f_{m_{1,i},e_1} - f_{m_{2,j},e_2}$, but not set cardinality as is the case elsewhere in the paper. Our intuition is that two MeSH terms have a higher strength (θ value) if they have *similar* weights (low absolute value of the difference between weight values) for their respective researcher. In other words, the similarity between two pieces of expertise, represented by MeSH terms, is adjusted by the difference in expertise level between the two researchers on these two pieces of expertise.

6. CONCLUSIONS

Accurate representation of researchers' expertise leads to more effective facilitation of research collaboration and more informed research planning and decision making. However, it is a very challenging task for a large institution and a diverse discipline.

In this paper, for the vast life sciences domain, we present a framework for representing and matching researchers' expertise based on the Medical Subject Headings (MeSH) taxonomy. Our main contributions are two-fold.

- A novel hybrid MeSH classification framework that combines three widely-used methods. Our evaluation demonstrates superior performance over state-of-the-art systems for MeSH classification using the entire MeSH taxonomy on a recent benchmark dataset.

- A novel researcher expertise matching algorithm based on semantic similarity from aggregated MeSH terms, exploiting structural information of the MeSH taxonomy.

Our proposed framework is domain-independent. Although we focus on MeSH and the life sciences in this paper, this framework can be applied to any broad scientific discipline that has developed a shared, comprehensive taxonomy/ontology for describing disciplinary topics. These disciplines include Computer Science, Engineering, Mathematics and Physics among others.

We plan a number of future work directions. We will continue to enhance our MeSH classification algorithm by exploiting the associations between MeSH terms and articles with the aim of uncovering latent MeSH labels. We will further investigate the researcher matching algorithm to exploit not only the structure of MeSH, but also its statistical properties, such as co-occurrences of MeSH terms in a dataset, and conduct user studies to evaluate its accuracy. We will also develop a prototype system that incorporates these functionalities.

7. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749, 2005.

[2] B. Aljaber, D. Martinez, N. Stokes, and J. Bailey. Improving MeSH classification of biomedical articles using citation contexts. *Journal of Biomedical Informatics*, 44(5):881–896, 2011.

[3] A. Aronson, J. Mork, C. Gay, S. Humphrey, and W. Rogers. The NLM Indexing Initiative's Medical Text Indexer. *Medinfo*, 11(1):268–272, 2004.

[4] A. Bellogín. Performance prediction in recommender systems: application to the dynamic optimisation of aggregative methods. Master's thesis, Escuela Politécnica Superior, Universidad Autónoma de Madrid, Madrid, Spain, July 2009.

[5] L. Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, Oct. 2001.

[6] R. Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, Nov. 2002.

[7] M. Cobo, A. López-Herrera, E. Herrera-Viedma, and F. Herrera. Science mapping software tools: Review, analysis, and cooperative study among tools. *Journal of the American Society for Information Science and Technology*, 62(7):1382–1402, 2011.

[8] J. Cohen. *Statistical Power Analysis for the Behavioral Sciences (2nd Edition)*. 1988.

[9] H. Fang and C. Zhai. Probabilistic models for expert finding. In *Proceedings of the 29th European Conference on IR Research*, ECIR'07, pages 418–430, Berlin, Heidelberg, 2007. Springer-Verlag.

[10] P. Ganesan, H. Garcia-Molina, and J. Widom. Exploiting hierarchical domain structure to compute similarity. *ACM Trans. Inf. Syst.*, 21(1):64–93, Jan. 2003.

[11] R. Genuer, J.-M. Poggi, and C. Tuleau-Malot. Variable selection using random forests. *Pattern Recogn. Lett.*, 31(14):2225–2236, Oct. 2010.

[12] M. Huang, A. Névéol, and Z. Lu. Recommending MeSH terms for annotating biomedical articles. *Journal of the American Medical Informatics Association*, 18(5):660–667, 2011.

[13] A. Jimeno-Yepes, L. Plaza, J. Mork, A. Aronson, and A. Díaz. MeSH indexing based on automatically generated summaries. *BMC Bioinformatics*, page 208, 2013.

[14] Y.-B. Kang, A. Zaslavsky, S. Krishnaswamy, and C. Bartolini. A knowledge-rich similarity measure for improving it incident resolution process. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, SAC '10, pages 1781–1788, New York, NY, USA, 2010. ACM.

[15] R. Kavuluru and Z. He. Unsupervised Medical Subject Heading Assignment Using Output Label Co-occurrence Statistics and Semantic Predications. In *Natural Language Processing and Information Systems*. 2013.

[16] D. Lindberg, B. Humphreys, and A. McCray. The unified medical language system. *Methods of Information in Medicine*, 32(4):281–291, 1993.

[17] S. A. Morris and B. Van der Veer Martens. Mapping research specialties. *Annual Review of Information Science and Technology*, 42(1):213–295, 2008.

[18] T. Pedersen, S. V. S. Pakhomov, S. Patwardhan, and C. G. Chute. Measures of semantic similarity and relatedness in the biomedical domain. *J. of Biomedical Informatics*, 40(3):288–299, June 2007.

[19] A. Porter and I. Rafols. Is science becoming more interdisciplinary? Measuring and mapping six research fields over time. *Scientometrics*, 81(3):719–745, Dec. 2009.

[20] F. Provost. Machine learning from imbalanced data sets 101. *Proceedings of the AAAI-2000 Workshop on Imbalanced Data Sets*, 2000.

[21] P. Resnik. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130, 1999.

[22] T. C. Rindflesch and M. Fiszman. The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text. *J. of Biomedical Informatics*, 36(6):462 – 477, 2003.

[23] G. Salton, editor. *Automatic text processing*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1988.

[24] N. Seco, T. Veale, and J. Hayes. An Intrinsic Information Content Metric for Semantic Similarity in WordNet. In *Proceedings of European Conference on Artificial Intelligence*, pages 1089–1090, 2004.

[25] W. Tang, J. Tang, T. Lei, C. Tan, B. Gao, and T. Li. On optimization of expertise matching with various constraints. *Neurocomputing*, 76(1):71 – 83, 2012.

[26] G. Tsoumakas, I. Katakis, and I. Vlahavas. Random k-labelsets for multilabel classification. *IEEE Trans. on Knowl. and Data Eng.*, 23(7):1079–1089, July 2011.

[27] V. Vasuki and T. Cohen. Reflective random indexing for semi-automatic indexing of the biomedical literature. *Journal of Biomedical Informatics*, 43(5):694–700, Oct. 2010.

[28] M. Wahle, D. Widdows, J. R. Herskovic, E. V. Bernstam, and T. Cohen. Deterministic Binary Vectors for Efficient Automated Indexing of MEDLINE/PubMed Abstracts. *Proceedings of AMIA Symposium*, 2012:940–949, 2012.

[29] X. Wei and W. B. Croft. Lda-based document models for ad-hoc retrieval. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 178–185, New York, NY, USA, 2006. ACM.