CAFE – Collaboration Aimed at Finding Experts

Neil Rubens, Dain Kaplan*, Mikko Vilenius, Toshio Okamoto

Graduate School of Information Systems,
University of Electro-Communications,
Tokyo, Japan
{rubens, mikko, okamoto} @ ai.is.uec.ac.jp

* Department of Computer Science,
Tokyo Institute of Technology,
Tokyo, Japan
dain@cl.cs.titech.ac.jp

Abstract

Research-oriented tasks continue to become more complex, requiring more collaboration between experts. Historically, research has focused on either finding a single expert for a specific task (expertise finding, or EF), or trying to form a group that satisfies various conditions (group formation, or GF). EF is typically group context agnostic, while GF requires complex models that are difficult to automate. This paper focuses on the union of these two, forming groups of experts. We concentrate in this paper on the expertise aspect of group formation, since without the needed expertise, regardless of other factors, the task can not be accomplished. Our proposed model, CAFE (Collaboration Aimed at Finding Experts), is a data-driven approach, easy to construct and dynamic with respect to the data. More specifically, we address the problem of finding a group of experts for a given task (research paper) by utilizing the data inherent to citation graphs.

Keywords automatic group formation, expertise finding, computer-supported collaborative learning (CSCL), informal learning, data mining, machine learning, link analysis

[PREPRINT] Please cite as:

N. Rubens, D. Kaplan, M. Vilenius, and T. Okamoto, “CAFE: Collaboration Aimed at Finding Experts,” *International Journal of Knowledge and Web Intelligence (IJKWI)*, vol. 1, iss. 3/4, pp. 169-186, 2010.

1 Introduction

In today’s knowledge-based economy, having the proper expertise is crucial to resolving a given task. However, nowadays work is rarely done in total isolation; projects often span multiple disciplines, and the disciplines themselves are growing more and more complex. Thus, it is not just about *finding* the right expertise, but about finding the right *set* of expertise in a collaborative setting. Historically, research has focused on either of these two tasks, namely, finding a single expert for a specific task (expertise finding), *or* constructing a group with members that best satisfy a manually created model representing what is needed (group formation). Expertise finding (EF) is limited in that it does not consider the collaborative setting, and group formation (GF) in that it is not fully automated, and requires the creation of complex models with well defined constraints and conditions.

We focus in this paper on the union of these two tasks. Further, we aim at automatically creating a group of experts best suited for a certain problem. We posit that regardless of other conditions, if a member of a group is not an expert their contribution will be limited (if present at all). We therefore ignore the other conditions present in GF by attempting to approximate the expertise of an individual in relation to the group. Our realm of interest is also limited in this paper to research oriented settings. We next outline some common use cases for the application of our proposal. After this we summarize EF and GF before entering an explanation of our research and experimental results. We end with a conclusion and future work.

1.1 Use Cases

In research oriented settings, there are many potential benefits of experts working in collaboration, including knowledge diffusion through sharing of ideas, exposure to different ways of thinking, providing a sense of community, and as a result increased motivation, etc. [35]. Simply speaking, we are connecting tasks and people. Below are some common scenarios in which having a group of experts would be beneficial.

Collaborative Research As research becomes more interdisciplinary and more intricate, the amount of collaborative research will continue to grow. In this case, the end goal is to produce a work of research. Therefore it is a matter of finding the right group of members with the required expertise (Figure 1a).

Figure 1: Collaborative Scenarios. Dashed lines will be determined by the group formation model.

Collaborative Assessment Research must often be assessed by peers to determine its quality; this occurs during the peer review process for papers submitted to conferences and journals, or when appraising grant applications. In this case both the documents to be assessed and the potential members may be fixed (e.g., a review committee); it is a matter of best arranging them to yield the most meaningful review (Figure 1b).

Collaborative Learning Let us consider the task of assigning the reading and presenting of papers in a graduate level course. Since we are limited to the students in the class, the potential members of the group are fixed. Often the papers that will be presented (or at least the topics) will be fixed beforehand by the supervising professor. The goal, then, is to assign papers (or topics) to the students in a way best matching their backgrounds and/or interests (Figure 1c).

1.2 Motivation and Contribution

As we have shown in the introduction, there is a gap between expert finding (EF) and group formation (GF) that we wish to fill. GF has been rather extensively studied in fields such as education (especially in the area of computer supported collaborative learning [29]), business, and psychology; many models have also been proposed [4, 32, 29, 8]. However, much less work exists on research-oriented settings (where the primary task is to perform research); further, GF models traditionally have a heavy reliance on the availability of concrete representations of members and tasks, e.g. prior knowledge of what is required from group members to accomplish the given task, and thus also details about that task. In many practical settings, however, such extensive knowledge may not be available [21]; and generally, the creation of such knowledge for each entity (member

or task) is labor intensive, meaning such a solution may not scale well. If requirements change, the assumptions made during the creation of the model may no longer hold, invalidating the model and requiring further human effort to correct. An inexpensive, automatic means for group formation, therefore, has tremendous appeal.

By recasting the GF problem as one of finding suitable experts for a given task, we can remove the complex model conditions needed for formulating GF. We posit that expertise is the primary factor in group formation; while other factors may be important, without the needed expertise the task can not be solved. So we take a look at expert finding (EF) to see how it can remedy this. However, EF’s primary goal is find the most suitable expert given a set of requirements for some task; finding a single expert is treated as the end-all solution. In other words, it treats finding an expert as an independent task and is entirely agnostic to the collaborative context.

This research aims at unifying these two tasks by proposing an EF method with regards to the group context, which we call the Collaboration Aimed at Finding Experts (CAFE) model. Our task is then to determine what expertise is required for accomplishing a given task, and then assess the fitness of the experts in this group context. As stated, we want to reduce the overhead for GF to make it scalable. For this, we propose a data-driven model that works as follows: first, data about learners and learning materials is obtained from existing data sources; then this data is preprocessed (linked into an interconnected network); machine learning methods are utilized to determine which features (i.e. patterns in the data) lead to a productive group; and lastly, these learned features are used to formulate a GF model dynamically. This also differs from traditional GF models, in which models are constructed using predefined criteria.

2 Related Works

As this research presents one possible solution for the synthesis of group formation (GF) and expertise finding (EF), these two fields are summarized below.

2.1 Group Formation

Group formation (GF) has been rather extensively studied in many fields, including psychology, sociology, business, and education. However, as the focus is generally not on automatically formulating a group of expertise related to a certain task, direct comparison is difficult. Computer Supported Collaborative Learning (CSCL) is probably the nearest match from within these fields and will be the focus for the remainder of this section; for this comparison, we can consider collaborative research (our aim) as a form of collaborative learning where the objective is to produce a novel work, or to evaluate the work of others. More generally speaking, collaborative research is a kind of collaborative activity. Collaborative activities include a variety of activities where two or more researchers work together towards a common goal. In addition, each re-

searcher may also have individual agenda, e.g. acquiring specific skills, arguing his/her point of view, etc. Basically, it can be said that collaboration comes in a large variety of different forms, from small tasks to processes that may span generations, from two people discussing, up to a whole society working together [7].

However, collaboration is by no means trivial. Aspects such as group dynamics, roles of participants in the collaboration, etc. have a substantial impact on the activity as a whole. For this end, a number of methods have been developed for forming collaborative groups automatically in an e-Learning environment (though their focus is on factors other than expertise).

One method [11] selects learners (members) of a group by maximizing its heterogeneity, where *heterogeneity* is defined by the personal traits of a learner. Another [32] uses learners' relative progress in course material as a criterion for group formation; when a suitable number of students reach a "point of cooperation," a collaborative activity is automatically triggered. The members in such a collaborative group – selected from the students that have reached the point of cooperation – are then decided automatically by the system or possibly intuitively by an administrator or a teacher. "Opportunistic group forming" [16] is fundamentally similar to this approach. However, instead of predefined points for collaboration, the system decides when a learner is in need of a collaborative activity (e.g. has trouble understanding a certain part in the course) and assigns roles to other learners based on their advancement/success in the learning material. In other words, other learners might be recommended for collaboration if they can help a learner in trouble (in a tutor or mentor kind of role), or if they have problems with the same learning objective as the person, for whom the collaboration was originally initiated. Users can also be automatically organized into e-learning communities based on their personal achievement, such as taking part in specific courses, submitting questions and assessments, etc. [37]. The aim of these methods is quite specific, focusing on factors other than expertise.

2.2 Expertise Finding

It is clear why experts are important: they can contribute their extensive knowledge to a variety of tasks, such as educating others, solving difficult problems, or assessing and guiding the research directions of others. The most traditional approach to expertise finding is typically a slow and burdensome process, involving directly contacting the individuals that are familiar with the areas for which expertise is required, and then relying on their ability to provide appropriate referrals.

Computers have mitigated this burden to a considerable degree. Several excellent surveys exist concerning this, such as [39, 23]. As a result of the aid of computers, expert finding systems (EFS) have started to gain acceptance and are being deployed in a variety of areas. The Taiwanese National Science Council utilizes EFS to find reviewers for grant proposals [38]; Australia's Department of Defense has deployed a prototype EFS to better utilize and manage its human

resources [27]; ResearchScorecard Inc.’s EFS allows a user to find and rank scientists involved in biomedical research at Stanford University and at the University of California in San Francisco. There are also several expertise finding platforms that are applicable to wider domains and are utilized by an increasing number of companies [23]. Further, many methods have been developed to automate the task of expertise finding, including language and topic modeling [38], latent semantic indexing [22], probabilistic modeling [2], and link analysis [17].

However, these methods are agnostic to the group context; this means that complex projects requiring many experts still lack the necessary tools to automatically select a suitable team.

3 Proposed Approach

3.1 Motivation

3.1.1 Group Formation

Traditionally group formation (GF) models are constructed from the data to represent each of the underlying entities, e.g. the task description and candidate profiles. The group is then formed by matching candidates to the task. With the appropriate data to construct the models from, and the necessary effort to create the models, this approach can yield good results. Construction of the underlying models from available data may in practice, however, be difficult. Further, constructing models often requires the consideration of many factors, e.g. group cohesiveness, roles/relationships, acuity, thinking and learning styles, etc. Automatic estimation of these factors from available data could be extremely difficult if not impossible. In our case, we use a collection of research papers, so such estimation is hardly feasible.

3.1.2 Expertise Finding

Methods utilized by expertise finding (EF), while considering the expertise of potential members (which we posit as being crucial to group formation), do not address the group context. Expertise finding, in fact, tends to be treated as an independent task, i.e. given a set of requirements, find an expert. However, in our settings, once the expert is located s/he will not work in isolation, but rather as a member of a larger group. In addition, since the task is assigned to a group, it may require expertise in a variety of unrelated knowledge areas (e.g. cardiovascular diseases and pattern recognition). Expertise finding tends to try to satisfy all of the expertise requirements in a single candidate. In the collaborative setting this may be impossible, undesirable, or produce an unsatisfactory result of selecting a candidate with only a limited knowledge in all of the required areas.

Figure 2: Group formation task (Section 3.2). Squares of different colors represent knowledge from different areas that the task deals with and that the researchers possess expertise in. Researchers in a group should collectively possess the knowledge needed to accomplish the task. For example, the researcher in the middle is not able to contribute knowledge from the needed areas, and therefore is not selected as a member of this group. (Note: in actual settings the number of researchers and tasks could be very large.)

3.2 Problem Formulation

We address the difficulty of model construction with traditional group formation (GF) models (Section 3.1.1) by concentrating on the expertise factor. We formulate the task of group formation in the following manner (Figure 2). We assume that there exists a description of the task at hand. In a research-oriented collaborative setting, the task description could correspond to a research proposal or an academic paper. The goal then is to identify experts that collectively possess the expertise required to accomplish the indicated task. Many experts are likely required, as the task likely requires expertise in a number of areas, e.g. data mining, e-Learning, and natural language processing. In this research we focus our efforts on extracting this information from a collection of research papers, containing authors, affiliations, and links. Such information is readily available in abundance, which makes it ideal for this task.

Generalized Assignment Problem We can recast the GF problem as a special case of the Generalized Assignment Problem (GAP) [10]. The objective is then, given a paper p (e.g. a task description), to choose a group of experts $M \subset \mathcal{M}$ that collectively possesses the most expertise (referred to as ‘reward’ in GAP) about p , i.e. $R(M, p)$. Traditionally, this task is formulated for constructing a group that maximizes the sum of rewards of its members $m \in M$ (where the group is of fixed size s):

$$\text{maximize} \quad R(M, p) = \sum_{m \in M} r(m, p) \quad (1)$$

$$\text{subject to} \quad |M| = s \quad (2)$$

However, in our settings rewards (expertise) is not necessarily additive. In some cases where expertise overlaps (several members possess expertise in the same area), the overlapping expertise becomes redundant and should not be fully rewarded. In other cases, some overlap/redundancy in expertise could be beneficial and should therefore be rewarded, e.g. collaboration may be difficult if there is no common knowledge base.

$$R(M, p) \neq \sum_{m \in M} r(m, p). \quad (3)$$

Expertise finding focuses on estimating r (the expertise of a single expert). As pointed out above, since the values of r may not be additive, it becomes difficult to estimate the overall group reward R based on the rewards of its members; more specifically, it is difficult to both determine the degree of expertise overlap and to quantify its overall effects on R . This makes reusing any existing methods difficult as well. We can, however, bypass the problematic estimation of R using r entirely, by foregoing estimation of r (and its interactions) and directly maximizing our estimation of R . This is the focus of this research.

Limitations This formulation has its limits by measuring only already existing expertise, and not the expertise that can be acquired (e.g. when a researcher starts a new research topic). The focus of our research is on utilizing existing expertise (as it is concretized), and not on expertise potential (which is anyways difficult to quantify); addressing this limitation is therefore beyond the scope of this paper.

Computational complexity, without any optimizations, may also be high, since all possible combinations of experts should be considered for maximizing R . It is possible to reduce computational complexity by applying existing algorithms designed for GAP [10] to produce a set of candidate solutions (note that these algorithms may not produce the optimal solution due to the additive difficulties caused by the interactions between values of r , but can nevertheless produce a list of candidate solutions that may contain the optimal one); we can next apply our algorithm to the candidate solutions to estimate R for each, and select the one with the highest score. As a concrete example, we can reduce computational complexity by discarding the “so called” experts that receive a low value for r ; the intuition is that they are not likely to contribute to the group because they are not an expert on the needed material. We can also immediately select candidate experts with a high value for r , as regardless of group constitution, they will likely contribute much.

Figure 3: Difficulty of constructing models from data (Section 3.3).

3.3 Modeling

Challenge The above problem formulation (Section 3.2) still faces the challenge of creating the model based on data. In our case, we need to identify knowledge areas required for accomplishing a given task (i.e. the area of required expertise), and to identify the corresponding knowledge areas from the profiles of researchers (Figure 3). The underlying data is often very complex. As we have chosen a collection of research papers for our data, the task description could be represented by a paper describing the task along with citations of related and utilized papers; expert profiles are also extracted from the collection of authored papers. Reducing all of this data into a simple set of knowledge areas is difficult.

Approach To address this challenge, we take a data-driven approach – letting the data to speak for itself [9, 13]. We represent the model directly by the data, without trying to reduce it to a model representation, and delay the reduction until the inference step in which data does provide some clues on the effectiveness of the reduction approach (Section 3.4).

A paper contains textual data (the paper’s content) and link data (citations, affiliations, authors, etc.). Processing textual data could be a complex and time consuming endeavor. For simplicity, we use only the link data. We represent the link data by a heterogeneous graph data-structure:

$$G = (V, E), \quad (4)$$

where V is a set of vertices/nodes, and E is a set of edges/links. Node types are: paper, person, publication venue (e.g. conference, journal), affiliation (e.g. uni-

Figure 4: Graph based representation of the models (Section 3.3).

Edge Type	Node Types	Directed	Semantics
wrote	person, paper	no	paper's author
cite	paper	yes	a paper cites another paper
published in	paper, publication venue	no	paper's publication venue
affiliation	person, affiliation	no	person's affiliation

Table 1: Edge Types.

versity, company). Edge types, shown in Table 1 include: wrote, cites, published in, affiliation.

3.4 Learning Scheme

For the prediction step, we strive to make only a few assumptions, creating a model based on the data. The data-driven approach may allow us to significantly reduce the time required for the implementation of the model (consisting mostly of implementing the machine learning algorithms) and provides for greater adaptability (i.e., as the underlying data changes, so does the model's behavior). In the following sections we describe details of the proposed approach.

At this stage, data can provide guidance on how to learn the model. That is, we can make an assumption that the paper's authors have expertise about their own paper. Note that we do not assume that the authors have the most expertise on their paper, e.g. editorial committee could easily possess more extensive expertise on the paper's subject. However, only authorship properties are contained within our data (there is no information related to editorials). Therefore we assume that the authors have a sufficient level of expertise on their

Figure 5: The relationship between the paper and group members.

paper; this logically follows in that they at least wrote their paper. We can use this assumption to carry out the task of learning the reward model of group expertise (Section 3.2) in a supervised manner. That is, given a paper p , we can assume that its authors M_p^* possess the maximum amount of expertise for p , i.e. $R(M_p^*, p) = 1$. On the other hand, we assume that randomly selected members will have little expertise on the paper, i.e. $R(M, p) = 0$, where $M \cap M_p^* = \emptyset$. We quantify partial matches as the ratio of correctly identified authors to total correct authors for p , e.g. if a paper has 3 authors, and 2 authors were correctly selected, then R would be $\frac{2}{3}$. More precisely, we define the group expertise reward function as:

$$R(M, p) = \frac{|M \cap M_p^*|}{|M_p^*|}. \quad (5)$$

We formulate the task of constructing a GF model as learning an approximation $\hat{R}$ of R , and then use it to predict which group possesses sufficient expertise for p .

To learn an approximation of R , we need training data; we obtain this data in the following manner. First, we randomly select a paper p ; we construct a pool of candidate authors by adding the actual authors to the pool along with other randomly selected authors. We then randomly construct permutations of authors from the candidate pool, insuring that each permutation has the same number of members in p , i.e. $|M_p^*|$, and calculate R for each.

We still need to decide how to represent the inputs to R , namely group members M and paper p . As discussed in Section 3.3, the underlying data is represented as the graph G . To obtain the data that relates the paper node with the nodes corresponding to the selected group members, we extract a subgraph $G' \subset G$ in the following manner. We start with the nodes corresponding to the paper and the group members and traverse them in a breadth first manner

Figure 6: By using features of subgraphs we can detect whether a group possesses sufficient expertise, e.g. the distance between group member nodes to the task description node should be small.

up to depth d . Our assumption is that the graphs for the groups that does posses the required expertise, will differ from those that do not (Figure 6). For example for the expert groups, the members could have cited the same (or similar) references, as the ones cited by the task description.

However, machine learning algorithms are primarily designed to work on numeric input and not on graphs [40]. Therefore, we need to represent the subgraph G' by a vector of features values denoted by $\mathbf{g}$, as described in Section 3.5. Note that by using only the features of the subgraph, instead of the graph, some information will be lost. Nonetheless, we assume that enough information is captured by the features of G' for learning a suitable predictive model.

3.5 Features

We try to use features that may represent important properties of the graph in relation to our task. In this section we briefly describe the features used and the intuitions behind using them.

Since some of the features (e.g. shortest path) are calculated in relation to

a pair of nodes (a source node and a destination node). For the source node, we use a node that represents the task description. For the destination node, we add a new node to the graph that represents a group, connected to the candidates, which allows us to use a single point that represents the entire group, rather than trying to aggregate their individual feature metrics (Figure 5).

Shortest Path Shortest path between the task description and a person may indicate that the person is familiar with the matters covered by the task, e.g. if both the task description and a person cite the same paper.

Average Path Using the shortest path alone may not be enough, since it is also conceivable that the shortest path could be due to coincidence, e.g. both papers citing the same funding source. The average path may provide a more complete idea of the relation between the nodes of interest.

Resistance Distance The resistance distance is equal to the resistance between two nodes on an electrical network [19]. The intuition behind this is that the denser the surrounding network is, the larger the resistance distance.

Centrality The centrality of a node measures the relative importance of the node within the graph. We use the common measures of network centrality: degree centrality, betweenness, closeness, and eigenvector centrality [34]. For example, a paper that cites many other papers may be less focused. On the other hand, a paper may be influential if it is cited by many other papers.

Graph Strength Graph strength could be used to compute partitions of sets of nodes and to detect zones of high concentration along edges. We use it as an indicator of strength of the relation between the task and the candidate members. As an alternative measure we also use a clustering coefficient that measures the degree to which nodes tend to cluster together [30] and also vertex connectivity [33].

3.5.1 Considerations

Feature Selection If the features that we selected are not relevant, then they may be disregarded by the machine learning algorithms. In cases where the underlying algorithm does not cope well with the presence of possibly multiple irrelevant features, we can employ a feature selection algorithm that selects a small subset of the most relevant features.

Granularity We first defined features based only on the task description node and potential member nodes. However, using these features alone we were not able to achieve good predictive performance (Section 4.1). A task could be considered to be represented by the papers that it cites (subtask nodes). However, the current set of features does not consider each of the subtasks individually,

Model Type	Implementation
Lazy model	k Nearest neighbors
Bayes model	Naive Bayes
Tree induction model	Random forest
Neural net model	Feed forward neural net
Function fitting model	Relevance vector machine
Logistic regression model	Kernel logistic regression
Support vector model	Support vector machine

Table 2: Predictive models used in the ensemble.

but rather in aggregate. Therefore, the final score could be dominated by only a few subtasks that make up only a portion of subtasks. We want to obtain a more wholistic picture of how expertise requirements for each subtask is satisfied. Therefore, in addition to the task-level features, we add the same features for subtasks. Doing this has allowed us to achieve much better performance (Section 4.2).

Machine learning algorithms require a fixed number of features; therefore we approximate the distribution of subtask-level features by the following percentiles: 0% (min), 25%, 50%, 75%, 100% (max).

Implementation We have utilized the following open-source network analysis frameworks to extract features of the graph: NetworkX [12], Java Universal Network/Graph Framework (JUNG) [25], the Statnet R package [15], and iGraph [6]. To speed up feature extraction from the graph we utilize approximations provided by the various network analysis packages.

3.6 Predictive Model

Ensemble Scheme Combining various predictive models in an ensemble manner has been shown to be effective in solving many complex problems [26, 20]. We use a bootstrap aggregating (bagging) scheme [3], where each model in the ensemble has an equal weight on predictions. If a more flexible way of combining predictors is need, we can use the stacking ensemble scheme [36] of training a master model that learns how to combine predictors. Bagging alone yielded sufficient accuracy, so we have chosen it for the ensemble scheme.

Ensemble Models To ensure a variety of models in the ensemble we have chosen to use models of different popular types where an open-source implementation is available for the machine learning frameworks utilized [24, 14]. We have selected the following predictive models (Table 2) : k nearest neighbors method (lazy model), Naive Bayes method (Bayes model), random forest (tree induction model), feed forward neural net with back-propagation (neural net model), relevance vector machine [31] (function fitting model), kernel logistic

regression [28, 18] (logistic regression model), support vector machine [5] (support vector modeling). We expect that a reasonably constructed ensemble of models will perform well on this task.

4 Experiments & Discussion

Since our model is data-driven, the settings of our experiments are influenced by available data. We have chosen to utilize the CiteSeer dataset [1], since it is one of the most comprehensive and openly available datasets of academic publications. The CiteSeer dataset contains data on 1.3×10^6 academic papers along with 26.5×10^6 corresponding citations that link the papers.

Our goal is to predict who has the needed expertise to accomplish the task at hand. In our settings, we consider writing an academic paper to be the task. The paper’s authors are then considered to be the experts who are able to accomplish the task. Note, that we do not assume that the authors have the most expertise on their paper, e.g. editorial committee could easily possess more extensive expertise on the paper’s subject. However, only authorship properties are contained within our data (there is no information related to editorials). Therefore we assume that the authors have a sufficient level of expertise on their paper; this logically follows in that they at least wrote their paper. Our task is then, given a paper, to predict who the paper’s authors are. More precisely, we use the features of the graph (Section 3.5) that relate the paper and its potential authors to make our prediction (Section 3).

We construct the training and testing data as described in Section 3.4. We randomly select 1,000 articles. For each article we do the following. We fix the size of the candidate author pool at 100, containing the real authors, and then randomly selected ones. We then create 20 sets of authors randomly selected from the pool along with one set of actual authors (all sets are of equal size). Data from one half of the randomly selected articles is used to train the model as described in Section 3.4. The other half is used to evaluate the model. For each trial the model’s predictive accuracy is measured by the mean absolute error (valued between 0 and 1 inclusively). As the baseline, we use the method that assigns expertise score using the uniform distribution.

In our experiments, we used only a small portion of all available articles and author pairs since extracting graph features is time consuming. We have performed several different training/testing data splits and obtained similar results; therefore we believe that the current number of points is adequate.

4.1 Task-level Features

In this experiment we investigate the accuracy of our model when only task-level features are used. That is we only consider the features of the graph that relate the paper to its candidate authors. For example, measuring graph strength indicates how strongly the paper’s node is connected to the candidate authors.

Features	Mean Absolute Error	Test Points Criteria (Author)
uniform (baseline)	0.533	any (all points are used)
task-level	0.729	any (all points are used)
subtask-level	0.264	any (all points are used)
subtask-level	0.103	no actual authors
subtask-level	0.126	all actual authors
subtask-level	0.391	same affiliation of paper’s authors
subtask-level	0.584	only one actual author

Table 3: Accuracy of the proposed approach utilizing different feature sets and under various settings. Error is measured by the mean absolute error (min value is 0, max value is 1). Test points criteria describe author-level conditions of the points that were included in the test set, e.g. “no actual authors” correspond to the pair (paper, candidate_authors), where none of the candidate_authors are the actual authors of the paper (in this case we expect model to output 0, meaning that none of the required experts are present among candidate_authors).

Somewhat unexpectedly we have achieved a rather high absolute error rate of 0.729 (worse than baseline method).

Discussion We examined results to try and find an explanation for unexpectedly bad results. We noticed that a small portion of subtasks (cited papers) often dominate many of the features. For example, shortest and longest paths being determined by a single node. Even an average path may be strongly influenced by a single paper node that is particularly far in the graph structure. However, each of the subtasks should have a contribution to a final score. Motivated by this we have introduced subtask-level features as described in Section 3.5.1 and this has allowed us to improve the accuracy significantly as described in the next section.

4.2 Subtask-level Features

After the subtask-level features were added to the predictive model (Section 3.5.1) error decreased by almost threefold, from 0.729 to 0.264. This indicates that considering features at the right level of granularity is very important.

Discussion First we examine the cases for which the model has achieved low error. In cases where all of the actual authors were given, the model was able to detect that all of the required expertise is in fact required, and was erroneous only in a few cases achieving a mean absolute error (MAE) of 0.126. An even lower error of 0.103 was achieved for the cases where no actual authors were among the candidates. Therefore the model can detect candidate authors that possess no or very little of the expertise that is required.

Interestingly, the error was higher (0.391) in the cases where the paper’s authors belonged to the same institution. We speculate that this is due to the

authors playing different roles from just providing expertise, such as providing supervision and/or direction for the research, as well as assistance in some technical matters not reflected by citations. Our model is only able to detect the expertise factor of the group, and therefore does not perform well in such a case. The cases where the error seemed to be the highest, namely MAE being 0.584, are the cases in which only one author was present (in the list of system selected candidates). This indicates that it is hard to gage the contribution of a single author (among several authors), since sometimes it could be disproportionately small or large. This may also imply that in addition to using the author ratio error metric, other metrics should also be used.

5 Conclusion & Future Work

As mentioned at the opening of this paper, in today’s knowledge-based economy being able to provide group expertise is becoming more crucial every day. The continuous expansion and increasing complexity of disciplines, and their growing overlap is evidence of this. Current methods of group formation (GF) and expert finding (EF) do not provide a good means to solve this; GF is often too labor intensive and difficult to automate, while EF remains agnostic of the group context.

This research proposed a method falling in the intersection of both these two methods’ interests (finding a group of experts). The proposed method can be thought of both as a method to take the group context into account for the task of expertise finding (since rather than trying to satisfy all the requirements with a single individual we aim at detecting when and which additional members are needed to maximize expertise), and as group formation based on expertise (since our focus is not on other factors often used in GF but on approximating a member’s expertise). Thus assessing a potential member’s expertise becomes crucial. Our presented model, *CAFE* (Collaboration Aimed at Finding Experts), is a data-driven approach to GF, easy to construct and dynamic with respect to the data. It is an ensemble-based predictive model. We showed the importance of defining the right features for representing a researcher’s expertise. Since assessing all candidates may not be feasible, we plan to address this issue in a future work.

References

- [1] CiteSeerX dataset. <http://citeseerx.ist.psu.edu>, 2009.
- [2] K. Balog, L. Azzopardi, and M. de Rijke. Formal models for expert finding in enterprise corpora. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, page 50. ACM, August 2006.
- [3] L. Breiman. Bagging predictors. volume 24, pages 123–140. Springer, 1996.

- [4] O'Malley C., editor. *Computer-supported collaborative learning*. Springer, 1994.
- [5] C.C. Chang and C.J. Lin. LIBSVM: a library for support vector machines, 2001.
- [6] G. Csardi and T. Nepusz. The igraph software package for complex network research. *InterJournal Complex Systems*, 1695, 2006.
- [7] Pierre Dillenbourg. *Collaborative learning: Cognitive and Computational Approaches*, chapter What do you mean by collaborative learning?, pages 1 – 19. Elsevier, Oxford, 1999.
- [8] Donelson R. Forsyth. *Group Dynamics*. Wadsworth Publishing, 5 edition, 2009.
- [9] Stuart Geman, Elie Bienenstock, and Ren Doursat. Neural networks and the bias/variance dilemma. *Neural Comput.*, 4(1):1–58, 1992.
- [10] E.S. Gottlieb and MR Rao. The generalized assignment problem: Valid inequalities and facets. *Mathematical Programming*, 46(1):31–52, 1990.
- [11] Sabine Graf and Rahel Bekele. Forming heterogeneous groups for intelligent collaborative learning systems with ant colony optimization. *Intelligent Tutoring Systems*, pages 217 – 226, 2006.
- [12] A. Hagberg, D. Schult, and P. Swart. NetworkX, High Productivity Software for Complex Networks. *Webová stránka <https://networkx.lanl.gov/wiki>*.
- [13] Alon Halevy, Peter Norvig, and Fernando Pereira. The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24(2):8–12, 2009.
- [14] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I.H. Witten. The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, 11(1), 2009.
- [15] M.S. Handcock, D.R. Hunter, C.T. Butts, S.M. Goodreau, and M. Morris. statnet: Software tools for the representation, visualization, analysis and simulation of network data. *Journal of statistical software*, 24(1):1548, 2008.
- [16] Mitsuru Ikeda, Junichi Toyoda, Riichiro Mizoguchi, Thepchai Supnithi, and Akiko Inaba. Learning goal ontology supported by learning theories for opportunistic group formation. *Artificial Intelligence in Education*, 1999.
- [17] M. Karimzadehgan, R.W. White, and M. Richardson. Enhancing expert finding using organizational hierarchies. In *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, pages 177–188. Springer, 2009.

- [18] S.S. Keerthi, K.B. Duan, S.K. Shevade, and A.N. Poo. A fast dual algorithm for kernel logistic regression. *Machine Learning*, 61(1):151–165, 2005.
- [19] DJ Klein and M. Randić. Resistance distance. *Journal of Mathematical Chemistry*, 12(1):81–95, 1993.
- [20] Y. Koren. The BellKor Solution to the Netflix Grand Prize. 2009.
- [21] D.W. Livingstone. Exploring the icebergs of adult learning: Findings of the first canadian survey of informal learning practices. *Canadian Journal for the Study of Adult Education*, 13(2):49–72, 1999.
- [22] KE Lochbaum and LA Streeter. Comparing and combining the effectiveness of latent semantic indexing and the ordinary vector space model for information retrieval. *Information Processing & Management*, 25(6):665–676, 1989.
- [23] M.T. Maybury. Expert finding systems. Technical report, MITRE Corporation, 2006.
- [24] Ingo Mierswa, Michael Wurst, Ralf Klinkenberg, Martin Scholz, and Timm Euler. Yale: Rapid prototyping for complex data mining tasks. In Lyle Ungar, Mark Craven, Dimitrios Gunopulos, and Tina Eliassi-Rad, editors, *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 935–940, New York, NY, USA, August 2006. ACM.
- [25] J. O'Madadhain, D. Fisher, P. Smyth, S. White, and Y.B. Boey. Analysis and visualization of network data using JUNG. *Journal of Statistical Software*, 10:1–35, 2005.
- [26] D. Opitz and R. Maclin. Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11(1):169–198, 1999.
- [27] P. Prekop. Supporting Knowledge and Expertise Finding within Australia's Defence Science and Technology Organisation. In *Hawaii International Conference on System Sciences*, volume 40, page 3236, Honolulu, Hawaii, January 2007. IEEE.
- [28] Stefan Rueping. myKLR - kernel logistic regression, 2009.
- [29] G. Stahl. *Group cognition: Computer support for building collaborative knowledge*. MIT Press, 2006.
- [30] D. Wagner T. Schank. Approximating clustering coefficient and transitivity. *Journal of Graph Algorithms and Applications*, 9(2), 2005.

- [31] M.E. Tipping and A. Faul. Fast marginal likelihood maximisation for sparse Bayesian models. In *Proceedings of the ninth international workshop on artificial intelligence and statistics*, volume 8, Key West, FL, USA, January 2003. Citeseer.
- [32] Martin Wessner and Hans-Rüdiger Pfister. Group formation in computer-supported collaborative learning. In *GROUP '01: Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*, pages 24–31, New York, NY, USA, September 2001. ACM.
- [33] D.R. White and F. Harary. The cohesiveness of blocks in social networks: Node connectivity and conditional density. *Sociological Methodology*, pages 305–359, 2001.
- [34] Wikipedia. Centrality — wikipedia, the free encyclopedia, 2009.
- [35] Wikipedia. Computer-supported collaborative learning — wikipedia, the free encyclopedia, 2009.
- [36] D.H. Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.
- [37] Fan Yang, Peng Han, Ruimin Shen, Bernd J. Kramer, and Xinwei Fan. Cooperative learning in self-organizing e-learner communities based on a multi- agents mechanism. In Tamás D. Gedeon and Lance Chun Che Fung, editors, *Australian Conference on Artificial Intelligence*, volume 2903, pages 490 – 500, Perth, Australia, December 2003. Lecture Notes in Computer Science.
- [38] Kai-Hsiang Yang, Tai-Liang Kuo, Hahn-Ming Lee, and Jan-Ming Ho. A reviewer recommendation system based on collaborative intelligence. *Web Intelligence and Intelligent Agent Technology, IEEE/WIC/ACM International Conference on*, 1:564–567, 2009.
- [39] D. Yimam-Seid and A. Kobsa. Expert finding systems for organizations: Problem and domain analysis and the demoir approach. *Sharing Expertise: Beyond Knowledge Management*. MIT Press, Cambridge, MA, 2003.
- [40] Xiaojin Zhu. *Semi-Supervised Learning with Graphs*. PhD thesis, Carnegie Mellon University, 2005.