AN ANALYSIS OF HELP SEEKING IN COURSE DISCUSSION FORUMS

BY

VYOM THAKKAR

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois Urbana-Champaign, 2022

Urbana, Illinois

Adviser:

Assistant Professor Suma Bhat

ABSTRACT

In the last few years, online learning has become increasingly prevalent in the student learning experience. With the advent of Massive Open Online Courses (MOOCs) and online courses at colleges and universities, course discussion forums have become an important medium for students to get help and ask questions. These discussion forums are also a medium to get to know other students and participate in class discussion activities.

The critical feature of large-scale course discussion forums is that, as the number of students learning online scales, the number of help-seeking questions scale, which can be met with the following options: (1) an increase in the number of course staff attending to these questions, (2) presence of an engaged community of students to actively help out each other or (3) to develop a technique in a way that these critical help-seeking discussion posts can be filtered out from the non-help-seeking posts in order to allow the course staff to deal with them in an efficient manner.

In this work, we used a discussion forum dataset from a chemistry course as the primary source of data for the investigations and experiments. We explore the use of Natural Language Processing (NLP) techniques in order to train models to classify a given text as help-seeking or non-help-seeking. We will explore the use of labeled text data from related domains to expand the primary dataset and experiment with transfer learning to improve classification performance.

We also performed a Social Network Analysis (SNA) to determine the correlation between the amount of student interaction on the discussion forums to the course outcome that they received. This work is followed up with a UI/UX exploration of integrating the findings from our work into the existing online course discussion forum experience.

To my parents, for their love and support.

ACKNOWLEDGMENTS

I would like to thank my advisor Dr. Suma Bhat for all the help and guidance provided. I learned a lot over these two years, and I will be forever grateful for this experience.

I would also like to thank Dr. Michelle Perry, Destiny Williams-Dobosz, Dr. Nigel Bosch, Renato Ferreira Leitao Azevedo, Amos Jeng and all of the people at The iLearn Group at UIUC without whom this work would not be possible.

Lastly, I want to take a moment to acknowledge my parents, brother, grandparents, uncles, aunts, teachers, friends, peers and all of the people who helped me become the person that I am today.

CONTENTS

LIST OF ABBREVIATIONS

API	Application Programming Interface
BERT	Bidirectional Encoder Representations from Transformers
DL	Deep Learning
GRU	Gated Recurrent Unit
HS	Help Seeking
IDF	Inverse Document Frequency
LR	Logistic Regression
LSTM	Long Short-Term Memory
ML	Machine Learning
MOOC	Massive Open Online Course
NB	Naive Bayes
NLP	Natural Language Processing
RNN	Recurrent Neural Network
SNA	Social Network Analysis
SVM	Support Vector Machine
TF	Term Frequency
TL	Transfer Learning
UIUC	University of Illinois Urbana-Champaign

Chapter 1

INTRODUCTION

1.1 Motivation

In the last few years, there has been a great surge in the number of online courses and the sheer number of people learning through these online courses. The COVID-19 pandemic, and the ensuing changes to people's lifestyles, have caused a vast number of universities to migrate their course delivery infrastructure to online [1]. Along with these changes, the pandemic has led to more people choosing to work and/or study from home, which has led to more free time in their lives, which would have otherwise been spent in commuting or other obligations.

Although the pandemic has greatly accelerated the growth of online learning, this trend has been rising for a long time. The rapid growth of online learning technology and online connectivity over the past decade has led to universal access to high-quality education materials irrespective of geographical location.

With for-profit (Coursera, edX) as well as non-for-profit efforts (MIT OCW, Khan Academy), there has been a rapid popularization of the concept of Massive Open Online Courses (MOOCs). MOOCs are large-scale online courses that anyone can enroll in. The space of MOOCs comprise of free-to-access as well as pay-to-access (one-time payment/subscription) courses.

The mode of delivery of content in MOOCs or other online courses is usually in the form of recorded video lectures or textual content like lecture notes or even a combination of these two different media. The content of the MOOC is often supplemented with assignments or exercises that the students need to take to strengthen their understanding of the underlying concepts taught. Furthermore, in order to facilitate discussion between the students taking the course and also to act as a means for seeking help from instructors or other students, online courses also have course discussion forums. These discussion forums also serve as a medium for instructors or other course staff to effectively convey relevant and important information to the students. This information can either be supplemental content or logistical information related to the everyday course affairs.

MOOCs by nature, have large numbers of enrolled students. This can often lead to difficulties in answering student questions in a timely manner. Because the number of course staff monitoring discussion forums is often much lower than the number of enrolled students, the burden of answering student questions cannot solely rest on the shoulders of the course staff. Engaged discussion forums where students help each other and answer questions of other students has been shown to lead to a better student experience [2]. However, motivating students to take time out and participate in the discussion forums is a challenge.

For whatever reason, if a course discussion forum is not engaged from a help-giving perspective, it can lead to increased burden on the course staff to answer questions. In such situations, it can be useful to figure out which discussion posts are help-seeking in nature so that course staff can give immediate attention to such posts. Since going through discussion posts and determining if they are help-seeking or not is often a time-consuming process, it is interesting to explore if such a filtering is feasible by leveraging Natural Language Processing (NLP) techniques.

In this thesis we will conduct all of our experiments on a dataset that we will refer to as the CHEM dataset, which consists of discussion posts from seven semesters of a chemistry course offering at UIUC.

From a text classification perspective, we also explore the use of transfer learning and techniques to expand the training corpus from similar data distributions.

In this thesis, we also look at determining if an increased participation in discussion forums leads to an improved student outcome in the course.

Lastly we explore the design of a UI/UX for course discussion forums, inspired by our findings in the thesis.

1.2 Problem Statement

1. To what extent can text classification techniques from the field of NLP be used to classify student posts as help-seeking or not help-seeking. We also look at techniques like transfer learning and training corpus expansion from similar data distributions, with the aim of improving classification performance.
2. To what extent does an increased participation in course discussion forums lead to a better student course outcome.
3. To design a UI/UX for course discussion forums inspired by the findings from our work.

1.3 Structure of the Thesis

Chapter 2 goes into the relevant theory behind the NLP, graph theory and Social Network Analysis (SNA) techniques that we will make use of in our work.

Chapter 3 addresses the problem of text classification of help-seeking for course discussion forum posts.

Chapter 4 addresses the problem statement of determining the influence of discussion forum participation in the course outcome for students.

Chapter 5 addresses the problem of designing a potential UI/UX for course discussion forums.

Chapter 6 is the conclusion and summarizes the findings from our work and addresses the possible future extensions of our work.

Chapter 2

THEORY

2.1 TF-IDF

Term Frequency - Inverse Document Frequency (TF-IDF) measures which words in a document are important (frequent) and unique (discriminative) [3].

Term Frequency (TF) simply measures the frequency of a word in a particular document. In order to compute the term frequency of a word w in a document d: $TF(w,d) = \frac{n[w,d]}{N}$, where $n[w,d]$ is the number of occurrences of a w in d, and N is the total number of words in a document (length of the document).

Inverse Document Frequency (IDF) measures the rarity of a particular word in the corpus. The IDF of a word w is computed as follows: $IDF(w) = \log \frac{N}{df(w)}$, where N is the total number of documents in the corpus and $df(w)$ represents the number of documents that contain the word w.

The TF-IDF of a word w in a document d is computed by multiplying the term frequency of the word in the document by the inverse document frequency of the word in the corpus.

This explanation for TF-IDF was compiled using the help of [3].

2.2 Naive Bayes

Naive Bayes is used to estimate the probability that a document belongs to a particular class. This estimation can be expressed as following, where $\hat{c}$ is the predicted class, d is a given document and C is the set of all possible classes: $\hat{c} = \text{argmax}_{c \in C} P(c|d)$. Using Bayes theorem we get that $P(c|d) = \frac{P(d|c)P(c)}{P(d)}$. Since, $P(d) = 1$, we can represent the predicted class computation as follows:

$$\hat{c} = \text{argmax}_{c \in C} P(c|d) = \text{argmax}_{c \in C} P(d|c)P(c)$$

Now, in order to estimate $P(d|c)$, we can make use of the bag-of-words model which assumes that each word in the document is independent of any other word occurrence. This means

that if a document d can be represented using the following bag of word representation $(w_1, w_2, ...w_n)$, then:

$$P(d|c) = P(w_1, w_2, ...w_n|c) = P(w_1|c) \cdot P(w_2|c) \cdot ... \cdot P(w_n|c)$$

This explanation for Naive Bayes was compiled using the help of [3].

2.3 Logistic Regression

Logistic regression (LR) is an approach for classification problems where the output is binary. Logistic regression uses the sigmoid function in the hypothesis definition. The sigmoid function g can be expressed as follows:

$$g(z) = \frac{1}{1+\exp -z}$$

The sigmoid function is incorporated into the hypothesis definition as follows:

$$h_\theta(z) = g(\theta^T x) = \frac{1}{1+\exp -\theta^T x}$$

In the above formulation, θ represents the learned weights, and x represents an input feature. The derivative of a sigmoid is simple to compute and can be expressed as follows:

$$g^{'}(z) = g(z)(1 - g(z))$$

We can formulate the probability of binary classification output y as follows:

$$P(y = 1|x; \theta) = h_\theta(x)$$
$$P(y = 0|x; \theta) = 1 - P(y = 1|x; \theta) = 1 - h_\theta(x)$$

This can be expressed more concisely as follows:

$$p(y|x; \theta) = (h_\theta(x))^y(1 - h_\theta(x))^{1-y}$$

Assuming that examples are independent and identically distributed and we have m examples, we can express the loss function $L(\theta)$ as follows:

$$L(\theta) = \prod_{i=1}^{m}(h_\theta(x^{(i)}))^{y(i)}(1 - h_\theta(x^{(i)}))^{1-y^{(i)}}$$

We can choose to maximize the log likelihood instead, using the following formulation:

$$\sum_{i=1}^{m} y^{(i)} \log h_\theta(x) + (1 - y^{(i)}) \log (1 - h_\theta(x^{(i)}))$$

This explanation for logistic regression was compiled using the help of [4].

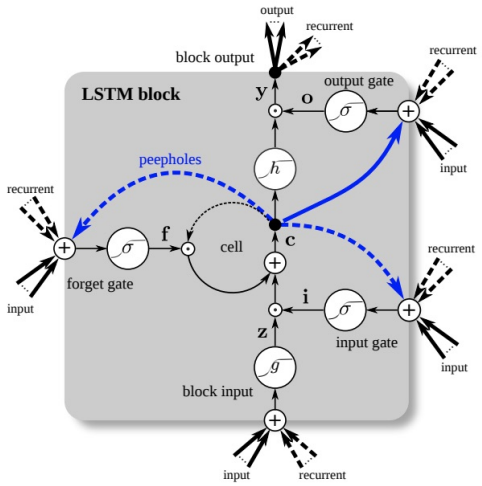

Figure 2.1: Diagrammatic representation of an LSTM cell (taken from [6])

2.4 Support Vector Machine (SVM)

Support vector machines try to find an N-dimensional hyperplane that tries to classify the data points using the largest possible margin. SVMs make use of the hinge loss which can be described as follows [5]:

$$l(y) = max(0, 1 - t \cdot y)$$

In the above formulation, t is the intended output, which can either be $+1$ or -1, and y is the output of the SVM classifier. Here, t, the model output can be represented as a function of the model parameter w and the input x as:

$$t = w^T x$$

Thus, if we have m training examples and n-dimensional feature space, the SVM objective that we need to minimize is the following:

$$min_w \frac{1}{2} \sum_{i=1}^{n} w_i^2 + \sum_{j=1}^{m} max(0, 1 - t_j \cdot y_j)$$

This explanation of SVMs was compiled using [4].

2.5 Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU)

LSTMs are a kind of Recurrent Neural Network (RNN) that is able to capture long-term dependencies in sequential data [6]. LSTMs maintain a memory cell and hidden state that

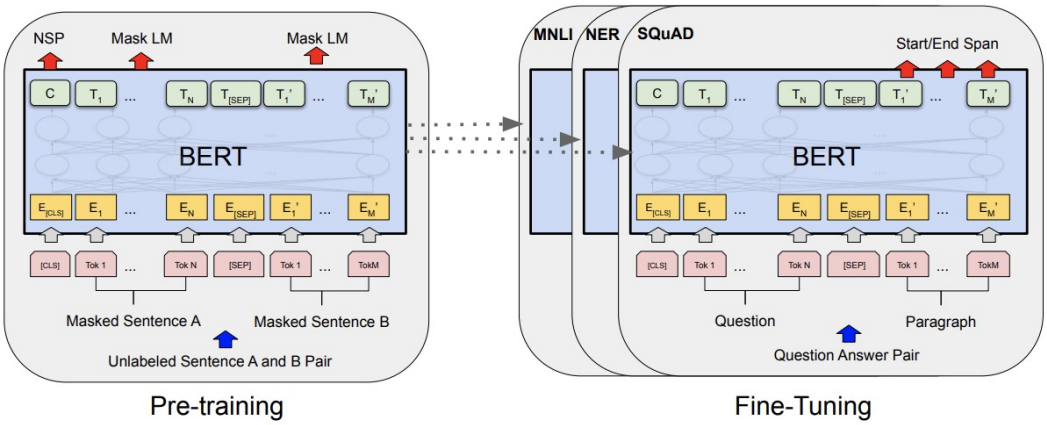

Figure 2.2: Structure of BERT architecture (taken from [10])

gets passed along the different LSTM cells [7]. LSTMs contain three gates (input, output and forget gates) that determine how much of the input and state should be preserved and passed on in the network [6]. Figure 2.1 demonstrates the structure of an LSTM cell.

LSTM cells are chained together in practice. LSTM chains can be bidirectional as well as unidirectional. A unidirectional LSTM only propagates information forwards, whereas bidirectional LSTMs propagate information in both directions. We will make use of bidirectional LSTMs in our implementation. Furthermore, we will use the many-to-one LSTM, wherein many LSTM cells are chained together to produce a single output. This explanation of LSTMs is compiled using [8].

GRUs are very similar to LSTMs, and can be considered as a simplification of the LSTM structure [9].

2.6 Bidirectional Encoder Representation from Transformers (BERT)

The BERT model architecture consists of a multi-layer bidirectional transformer encoder. BERT consists of two phases: pre-training and fine-tuning. The first phase refers to language model pre-training. This pre-training is done on an unlabeled dataset, which in the context of BERT is the BooksCorpus and English Wikipedia. Pre-training has been shown to improve the performance of NLP tasks. Then, once the model has been pre-trained, it is fine-tuned on downstream NLP tasks, which in the context of this thesis, would be text classification.

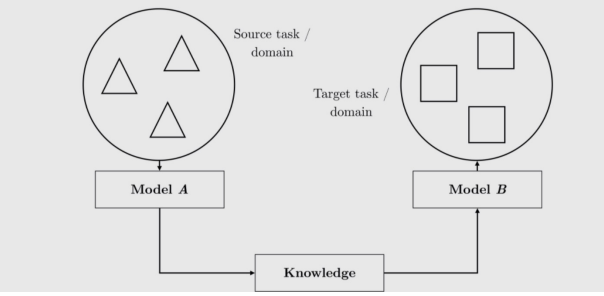

Figure 2.3: General transfer learning setting (taken from [11])

This explanation was compiled using [10]. Figure 2.2 is a diagrammatic representation of the BERT architecture.

2.7 Transfer Learning

Transfer Learning is a technique in which knowledge is extracted from a source setting, and applied to a different target setting [11]. The classical supervised learning setting is limited by the amount of labeled data that is available for training. Transfer learning allows us to deal with situations where enough labeled data is not available for the task that the model is trying to solve for. In particular, transfer learning leverages labeled data from a different, but related domain or task [11]. Figure 2.3 illustrates the general transfer learning setting.

Figure 2.4 illustrates the most popular transfer learning scenarios in NLP. Of these, domain adaptation and sequential transfer learning will be of particular interest to us.

Domain adaptation is used when the source and target settings solve for the same tasks, however, labeled data is only available for the source task. Furthermore, the source and target data also arise from different domains. In such a scenario, domain adaptation tries to bridge the gap between the difference in domains such that a model trained on the source setting can effectively generalize to the target setting [12].

Sequential transfer learning is becoming an increasingly popular technique in NLP. In this setting, labeled data is available for the target domain, and the source and target settings solve for different tasks. The source and target tasks are learned sequentially, such that the knowledge learned from the source setting helps in learning parameters for the target task. For example, in BERT, the source dataset consists of a large unlabeled corpus of text on which it is pretrained. Then, each downstream task represents a target task which the model is fine-tuned on [11].

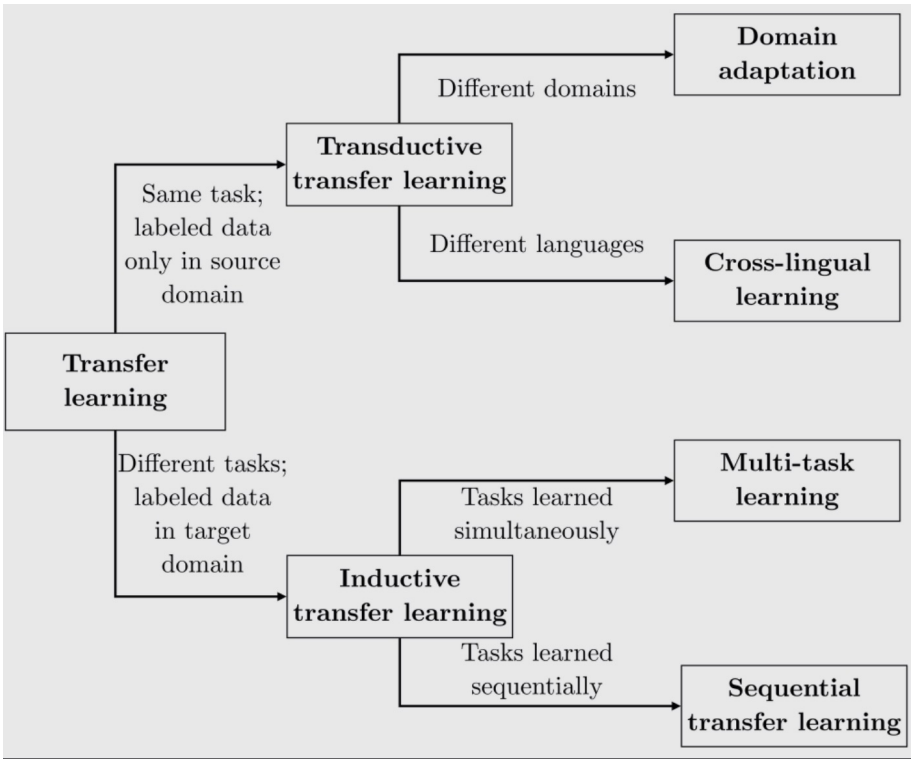

Figure 2.4: Types of transfer learning scenarios (taken from [11])

2.8 Graphs

Graphs can be represented using a set of nodes V and a set of edges E. Each edge connects two nodes in the graph. The nodes that are connected by an edge are considered neighbors.

When a graph is traversible in both directions, it is an undirected graph. On the other hand, a graph can be directed, which means that any edge between two nodes can only be traversed in one direction.

The degree of a node in a graph is mathematically equivalent to the number of edges that have that particular node as an endpoint.

The adjacency matrix of a graph is a square matrix (of size $|V| \times |V|$) which holds information about which pairs of nodes in a graph are neighbors (or adjacent). An element of the adjacency matrix ($a_{v,t}$) is 1 if v and t are neighbors and 0 otherwise [13].

This explanation for graphs was compiled using the help of [14].

2.9 Centrality

Centrality is a score given to nodes in a graph, based on how "important" or "influential" the node is with respect to the network.

Degree Centrality: This is one of the simplest measures of the centrality of a node. The degree centrality of a node is computed by counting the number of direct links (edges) shared with other nodes. [15].

Eigenvector Centrality: The eigenvector centrality score for a node i can be represented using c_i and expressed as follows:

$$c_i = \frac{1}{\lambda} \sum_{j \in G} a_{i,j} c_j$$

In the above equation, G is a graph, $a_{i,j}$ is the adjacency matrix element corresponding to the nodes i and j and λ is a constant. By rearranging the above equation we get the following where A is the adjacency matrix:

$$Ac = \lambda c$$

The eigenvector centrality is then the greatest eigenvalue obtained using the above equation.

This explanation for eigenvector centrality was compiled using [16].

Pagerank Centrality: The pagerank centrality of a node v can be represented as $PR(v)$ and expressed as follows:

$$PR(v) = (1 - c) + c \sum_{t \in Pnt_{in}(v)} \frac{PR(t)}{|Pnt_{out}(t)|}$$

In the above expression, c is a damping factor in the range [0,1], $Pnt_{in}(v)$ is the set of nodes pointing to v, and $Pnt_{out}(t)$ is the set of nodes pointed by t.

This explanation was compiled using [17].

Chapter 3

TEXT CLASSIFICATION OF HELP-SEEKING

3.1 Introduction

In this chapter of the thesis, we will be developing NLP models to classify a discussion forum post as help-seeking in nature or not. One of the fundamental questions that we intend to answer in this chapter is: To what extent can the help-seeking nature of discussion forum posts be accurately detected using popular natural language processing techniques in the field of text classification? Given the relatively small size of our initial labeled dataset (the CHEM dataset), we will also explore the use of transfer learning, techniques to expand this corpus and to what extent these techniques lead to improved performance. One of the most important criteria that we need to ensure when we expand our existing training corpus is that the added data comes from a similar distribution as the CHEM dataset. In our study, we identified the Stanford MOOCPosts dataset [18] and the r/HomeworkHelp subreddit (`https://www.reddit.com/r/HomeworkHelp/`) as possible data sources to expand our existing dataset.

On a broader level, the goal of this exploration is to inform the possibility of automatically filtering discussion posts on a learning forum as help-seeking or non-help seeking. This classification can direct the course staff's immediate efforts toward these urgent help-seeking posts, with the aim of improving the student experience by providing timely help. In Chapter 5, we will explore a direct application of the work in this chapter to a UI/UX for course discussion forums.

3.2 Technical Problem Statement

1. Experiment with classical machine learning (ML) models (naive bayes, logistic regression, SVM), RNN-based NLP models (GRU, LSTM) and pre-trained transformer model (BERT) to determine the best performing model on the HS text classification task on the CHEM dataset.

2. To identify similar-domain text corpora to the CHEM dataset and explore transfer learning and corpus expansion as techniques to potentially boost performance.

3.3 Technical Challenges

One of the central technical challenge that we face in our study is the limited quantity of textual data that our NLP models can learn from. One of the challenges of deploying NLP at scale is the lack of labeled data to tackle a given target NLP task. In such situations, there are two options, one is to label this data in-house or using a data annotation service like Amazon Mechanical Turk. The other option is to either utilize unlabeled data and adapt it to the given task or to make use of labeled data from a similar domain and use it for the target task. In our study, we want to explore the impact that using labeled data from a similar domain as the target task has on the performance of an NLP model on the target task.

3.4 Related Work

The scope of the work in this chapter falls at the intersection of the following domains:

Confusion Detection in Course Discussion Forums

With the rise of online learning, there has been a lot of work done in the field of confusion detection in forum posts.

Agrawal et al. [19] explore the classification of confusion in discussion forums by experimenting with logistic regression, SVM and naive bayes, three models that we will use in the experiments in this chapter. However, in our work we will also be exploring the use of RNN-based models as well as transformer-based models.

Geller et al. [20] found great success in using a pretrained BERT model on confusion classification on raw text data and found that this approach outperformed other classical ML models that made use of hand-designed features. In our work we will also be experimenting with using pretrained BERT to perform confusion classification on raw textual data from discussion forums.

Geller et al. [21] use student hashtags in the posts to detect confusion and also looked at automatically classifying confusion in posts that do not contain hashtags, by using logistic

regression.

Zeng et al. [22] make use of content-related lingustic features as well as community-related features to detect confusion in posts belonging to the Stanford MOOCPosts dataset. This work shows that their feature-engineering driven approach outperforms other available algorithms for confusion detection. Another important result that this work highlights is that predicting related sentiments like "Urgency" and "Confusion" in discussion posts are highly correlated, which means that training a classifier to predict one could effectively be used to make predictions on the other task. This is a significant finding, and we will be making use of this result in our work when applying transfer learning on models trained on a different distribution than the target dataset distribution.

Transfer Learning/Domain Adaptation for Confusion Detection in Course Discussion Forums

Bakharia et al. [23] pointed out that their study observed an inability for models trained on one MOOC to generalize and perform well on MOOC posts belonging to a different domain. This work also highlighted the need for future work in the space of transfer learning and domain adaption as possible ways to deal with this lack of generalization.

Brahman et al. [24] found that "Confusion" and "Urgency" text classification tasks on the Stanford MOOCPosts dataset are strongly correlated tasks. This shows that a text classification model trained on one task would be able to generalize well to the other task. This work also demonstrates that training on correlated tasks in a multi-task learning setup lead to an improvement in recall when compared to the single-task setup. This shows that a multi-task learning approach enabled their model to learn hidden abstractions that it would not have learned otherwise.

Zeng et al. [25] proposed an algorithm for expanding the training corpus using unlabeled data from a different MOOC by considering examples that are the most dissimilar to examples in the labeled dataset, but that the classifier has a high prediction confidence in. This work assumes a setting where there are two distributions of data: the source and target domains, where we have labeled data for the source domain but the target domain is the task that we want to optimize for, but which does not have a labeled dataset. The work found that their approach outperforms other domain adaptation models in the target domain. The work also found that domain adaption using their approach gave better performance than only using the labeled source dataset in training.

Wei et al. [26] apply transfer learning to the Stanford MOOCPosts dataset by evaluating the performance by training on one domain (subject matter) and evaluating on the other

domain. It is important to note that the Stanford MOOCPosts dataset consists of data from three different domain areas (humanities/sciences, medicine and education) [18].

3.5 Data

3.5.1 UIUC CHEM Dataset

This is the primary dataset that we will use for all of our experiments. This dataset was compiled and assembled by the iLearn group at UIUC. The dataset as a whole, contains discussion forum posts obtained from seven semesters of a UIUC chemistry course. The text data that we obtained from the discussion forums was anonymized such that the identity of the student is completely protected. Furthermore, the data was collected and was available for analysis only after the completion of the course and after the grades had been finalized [27].

Just like any internet forum, there are different levels for any given discussion to take place. We will be using the term "Level-0" posts to refer to discussion posts that are not made as a response to any other posts, but that are directly initiated by a student. On the other hand, we will use the term "Response-Level" posts to refer to posts that are either responses to "Level-0" posts or another "Response-Level" post.

However, for this chapter, we will restrict our experiments to only "Level-0" posts. This means that we will only seek to classify "Level-0" posts as either help-seeking or not.

The "Level-0" posts are either categorized as "not help-seeking" (label 0) or "help-seeking" (label 1). More specifically within the help-seeking category we have observed three distinct ways of seeking help [28]:

1. Straight questions
2. Implicit appeal to the community for help
3. Explicit appeal to the community for help

However, for this study we have made a simplification by unifying all of these different ways of help-seeking into a single category representing help-seeking intent.

3.5.2 Stanford MOOCPosts Dataset

The Stanford MOOCPosts dataset contains 29,604 anonymized student posts from eleven Stanford Public Online courses [18]. The reason that we decided to use this dataset to augment the UIUC CHEM dataset is that both come from roughly similar distributions, i.e

student posts from online course discussion forums. One of the main differences between these two datasets is that is our primary dataset consists of chemistry-related posts, however the Stanford MOOCPosts dataset includes data from courses in humanities/sciences, medicine and education [18]. Despite the differences in domain or subject matter, we expect the linguistic patterns of seeking help to be similar, which would in turn allow our NLP models to learn from a larger sample of data. The other critical difference is that unlike the UIUC CHEM Dataset, the Stanford MOOCPosts dataset does not label the posts as help-seeking or not help-seeking. However, this dataset has posts containing the "Confusion" category which is labeled on a scale of 1-7. To be more precise, the label "1" for confusion represents "not confused" and the label "7" represents "very confused". The labels "2-6" represent a continuously increasing spectrum of confusion. Although the Stanford MOOCPosts dataset contains other information (categories) like "Urgency", "Sentiment", "Opinion", "Answer", "Question" and other categories, we will only be using the "Confusion" label because we believe that it serves as the best proxy for capturing the sentiment of help-seeking. In the following section, we will describe how we convert the confusion labels to corresponding help-seeking labels.

3.5.3 r/HomeworkHelp Subreddit Posts

The r/HomeworkHelp is a subreddit where students ask help for their homework. We made use of PRAW (Python Reddit API Wrapper) to extract 1000 (maximum limit) posts from this subreddit. Since these posts ask for homework help, they are help-seeking by nature, thus, when augmenting to our primary dataset, these posts would be labeled as "help-seeking". The reason why we chose to make use of this data source is that these posts are made on a discussion forum, and it consists of students asking for help, thus we can assume that the distribution of posts is very similar to the distribution of help-seeking posts in the CHEM Dataset. Although the subject matter of these posts has a wide distribution of different topics, we believe that the underlying linguistic patterns of seeking help remain the same, thus we believe that augmenting additional information from these posts to our primary dataset would be valuable.

3.6 Help-Seeking Text Classification in CHEM Dataset

3.6.1 Structure of the CHEM Data

Table 3.1 contains a summary of the organization of the CHEM dataset in terms of the number of posts that were obtained from each of the different semesters.

In total, if we add up the number of posts from each of the semesters, we get a total count of 2753 posts across all of the seven semesters. After cleaning up the dataset and removing unlabeled posts, we get a total of 2668 labeled posts.

Out of the 2668 validly labeled discussion posts, we can see the breakdown between the number of help-seeking and non-help seeking post in Table 3.2.

3.6.2 NLP Approach

In order to perform text classification, we chose the following NLP algorithms: naive bayes, logistic regression, support vector machine, GRU, LSTM and BERT. Naive bayes, logistic regression and support vector machines are classical ML techniques, GRU and LSTM are Deep Learning (DL)-based RNN models, whereas BERT is a transformer-based model. We will have a slightly different pipeline when approaching classical, RNN-based models and transformer-based models, and we will demonstrate that further in the following sections.

3.6.3 Training and Validation

Since we have discussion posts from seven semesters, in our experimental evaluation, we train an NLP model using data from six semesters and evaluate its performance on the data from the unseen semester that was held out from the training set. Since different semesters have different number of posts, we come up with a single evaluation metric by averaging the

Table 3.1: Distribution of CHEM posts across semesters

Semester	**Number of posts**
Semester 1	348
Semester 2	375
Semester 3	1118
Semester 4	41
Semester 5	494
Semester 6	333
Semester 7	44

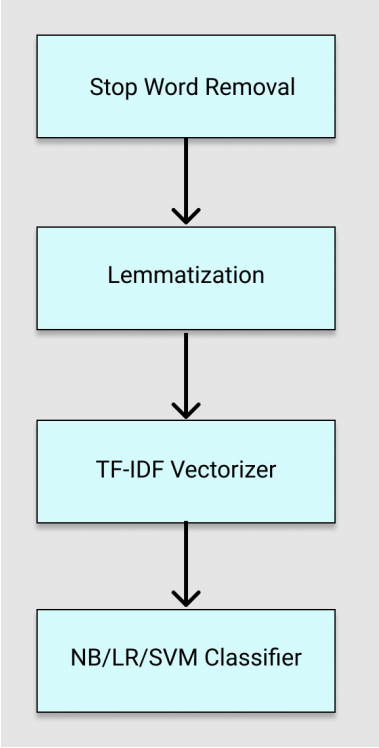

Figure 3.1: Flowchart of the NLP pipeline for classical ML models

validation performance from each of the semesters. This means that in order to evaluate the performance of an NLP model, we conduct seven runs in total, holding out each of the seven semesters once and training on the rest. We then compute the mean of the validation performances across all of the seven semesters that were evaluated.

The metric that we have chosen to compute the validation performance is the F1 score between the model predicted labels and ground-truth labels in the validation set. The F1 score is the harmonic mean of the precision and recall [3]. For reference, we will also display the validation accuracy, precision, recall as well as the F1 score when comparing the performance between the different models.

3.6.4 NLP Pipeline for Classical ML Models

The Figure 3.1 contains a flowchart of the NLP pipeline of our system.

Table 3.2: Distribution of help-seeking and non help-seeking posts in CHEM dataset

Number of Help-Seeking	**Number of Non Help-Seeking**
862	1806

Stop Word Removal

Stop words refer to the most common words in a language [3]. The reason why we remove stop words from our data is that because they are extremely common words. Thus, they do not add a unique distinguishable meaning to a given text. Some of the stop words in English include: “a”, “the”, etc.

Lemmatization

Lemmatization is a text normalization technique that replaces words with the root (base form) of the word (known as the lemma) [3].

TF-IDF Vectorizer

In order to vectorize (convert text into a numerical format that can be processed by the machine learning model) our discussion posts, we will be making use of the TF-IDF vectorization technique. The TF-IDF representation captures which words in a discussion post are unqiue and important to that post.

NB/LR/SVM Classifiers

A detailed explanation of the workings of these classifiers is provided in Chapter 2.

3.6.5 NLP Pipeline for RNN-Based Models

The Figure 3.2 contains a flowchart of the NLP pipeline of our system.

Tokenizer

In Deep Learning (DL) NLP models, the tokenizer is used to map each word to a unique token that can be processed by a neural network/DL model. We made use of the tokenizer module provided by Tensorflow API [29] in order to implement this block.

Padding

Because RNN models require uniform-sized inputs, we need to make sure the the output of the tokenizer is padded to ensure uniform length. We perform post-padding, i.e padding

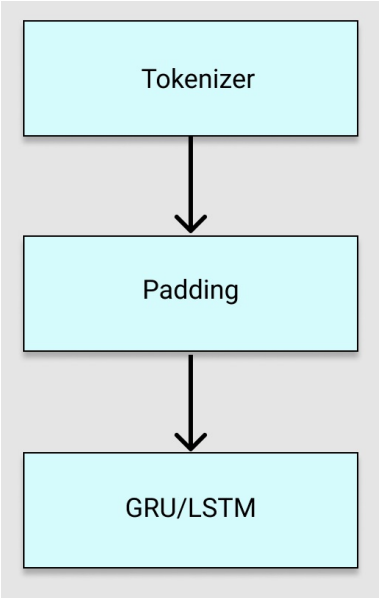

Figure 3.2: Flowchart of NLP pipeline for RNN-based models

after the end of token list in order to uniformly size. We made use of the pad_sequences module of the Tensorflow API in order to implement padding.

GRU

An explanation of the GRU is provided in Chapter 2. Figure 3.3 demonstrates the different layers in our GRU architecture. The first layer in the GRU is an embedding layer. The embedding layer converts a given word to a vector embedding representation. This embedding layer feeds into a bidirectional GRU layer which then feeds to a custom neural network consisting of a hidden layer and a single output neuron.

LSTM

An explanation of the LSTM is provided in Chapter 2. Figure 3.3 demonstrates the different layers in our LSTM architecture. The first layer in the LSTM is an embedding layer. This embedding layer feeds into a bidirectional LSTM layer which then feeds to a custom neural network consisting of a hidden layer and a single output neuron.

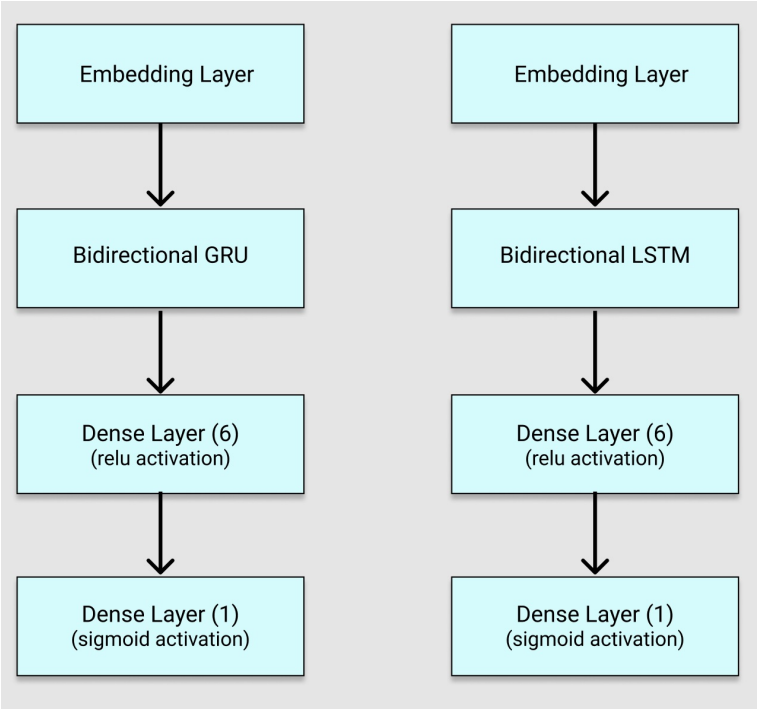

Figure 3.3: Architecture of GRU and LSTM

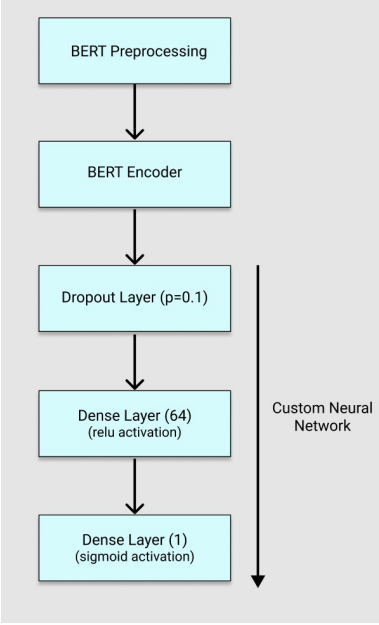

Figure 3.4: BERT pipeline

3.6.6 NLP Pipeline for BERT

The Figure 3.4 contains a flowchart of the BERT pipeline.

BERT Preprocessing

We made use of the BERT preprocesser for the text preprocessing step. This preprocessor uncases (converts text to lowercase) the input text and removes any accent markers. Then, it tokenizes the preprocessed text.

BERT Encoder

We made use of the BERT encoder as the transformer encoder. This BERT encoder consists of 12 transformer blocks, a hidden layer size of 768 and 12 attention heads. This BERT encoder is pretrained on Wikipedia and BooksCorpus [10].

Custom Neural Network

The output from the BERT encoder is then fed to our custom neural network. This neural network consists of a dropout layer (for regularization) and one hidden layer of size 64 which then connects to a single output layer neuron.

3.6.7 Results

Table 3.3 demonstrates the performance of the different NLP models. Although we include the training accuracy, validation accuracy, F1 score, precision and recall in the table, we will only use the F1 score as the single real number metric to evaluate and compare model performance. The other metrics are included only as a reference. Table 3.3 shows that we

Table 3.3: Performance of the models

Model	**Val. Acc.**	**F1 score**	**Precision**	**Recall**
Naive Bayes	0.856	0.775	0.709	0.894
Logistic Regression	0.926	0.865	0.911	0.829
Support Vector Machine	0.928	0.861	0.892	0.836
GRU	0.899	0.840	0.797	0.897
LSTM	0.905	0.858	0.835	0.886
BERT	0.927	0.860	0.891	0.833

obtained the best F1 score of 86.5% from the logistic regression model. The SVM comes behind logistic regression as a close second-best model with an 86.1% F1 score. We also get an 86.0% F1 score with BERT which is a close third-best model. Except for naive bayes, both the classical ML models had a better performance than RNN-based models. From a theoretical standpoint, this makes sense, because classical ML models have been shown to have better performance than DL models when the size of the training set is relatively smaller. Since BERT is pretrained on a large corpus of text and because of the recent advances in transformer-based architectures, the performance is justified. Moreover, in terms of validation accuracy, SVM had the best performance and BERT was a close second, with LR coming in third. Even for validation accuracy, the classical ML models except naive bayes and the transformer-based BERT showed better performance than RNN models. When we examine the precision score for the various models, we again observed that logistic regression, SVM and BERT have the highest scores, with LR being the best performing model. This means that whenever logistic regression classified a post as HS, we have a 91% confidence that the post is indeed HS. However, in the case of recall, we observe that GRU has the highest score. Naive bayes is a close second and LSTM is a close third. BERT, logistic regression and SVM have the lowest recalls. This means that for recall, we see that RNN-based approaches have success. For GRUs we can say that on average, it correctly identified 89.7% of all HS posts in the corpus. There are different situations in which recall or precision are more important, however, in this experiment we have assumed both to be equally important, thus, identifying the best model based on F1 score. In summary, since logistic regression has the best F1 score, it gives us the best performance in the HS text classification task.

3.7 Dataset Expansion and Transfer Learning

3.7.1 Help-Seeking Text Classification in CHEM Dataset Expanded Using Reddit Data

In this section we demonstrate the changes in performance, when the help-seeking posts from the r/HomeworkHelp subreddit are added to the training dataset comprising of CHEM posts. An important note to consider is that we only augment the training dataset with the Reddit data and the validation accuracy is only computed on the unseen CHEM data from a given semester. As was previously stated, the total number of posts that were obtained from r/HomeworkHelp subreddit is 1000, and all of these 1000 posts are help-seeking in nature.

Apart from expanding the training dataset, the rest of the NLP pipeline is the same.

Results

As can be seen from Table 3.4, expanding the training data with posts from Reddit decreased the F1 score of all of the NLP models. Correspondingly, the best performance obtained from training via this expanded dataset is an F1 score of 84% via logistic regression. BERT is the second-best performing model and SVM is the third. Even in this setting, the top-three performing models are the same. However, these models have a lower F1 score than the setting with no dataset expansion as was seen in the previous section.

This result indicates that expanding the training set with Reddit data does not improve the text classification performance. This also indicates that the CHEM dataset and Reddit data might not have distributions that are as similar as we expected.

3.7.2 Dataset Expansion Using Stanford MOOCPosts Data

In this section we demonstrate the changes in performance, when StanfordMOOCPosts are added to the training dataset comprising of CHEM posts. An important note to consider is that we only augment the training dataset with the StanfordMOOCPosts data and that validation performance is only computed on the unseen CHEM data from a given semester.

Apart from augmenting the training data, the rest of the NLP pipeline is the exact same as was described in the previous section.

Adapting Stanford MOOCPosts Data to the Help-Seeking Text Classification Task

The Stanford MOOCPosts dataset contains 29,604 student posts. From a help-seeking perspective, the "Confusion" label of a post is of particular interest to us. This column has labels that fall in the range of 1-7. A numerically higher label indicates a higher degree of confusion

Table 3.4: Performance of the models using dataset expansion from r/HomeworkHelp

Model	**Val. Acc.**	**F1 score**	**Precision**	**Recall**
Naive Bayes	0.824	0.745	0.670	0.883
Logistic Regression	0.909	0.840	0.832	0.857
Support Vector Machine	0.911	0.833	0.825	0.849
GRU	0.899	0.835	0.795	0.890
LSTM	0.894	0.831	0.808	0.872
BERT	0.915	0.836	0.824	0.858

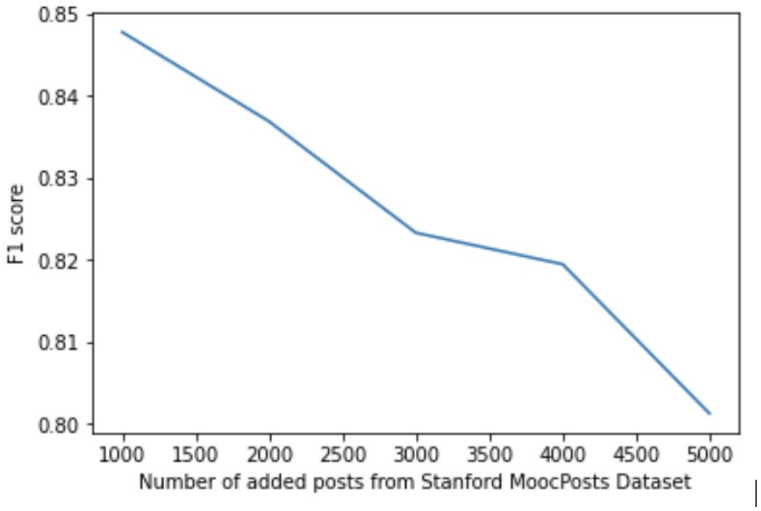

Figure 3.5: Logistic regression: change in F1 score performance based on number of Stanford MOOC posts added

in the post. If a post had a "Confusion" label ≥ 4.0, we gave it a "Help-Seeking" label of 1 and otherwise, the post was assigned a label 0 corresponding to "Not Help-Seeking".

Since the Stanford MOOCPosts dataset has 29,604 posts and the CHEM Dataset has 2753 posts, using all 29,604 posts would mean that our models would try to fit the Stanford dataset more than we desire. It is important to note that one of the main reasons to introduce dataset expansion is so that it can learn from a larger pool of data, and generalize to unseen posts in a more robust fashion. However, by training the models with more examples from the additional dataset, we would be tuning it to perform better on a different distribution than the intended CHEM data. Thus, we also experiment by observing the performance changes as we augment with larger amounts of data as illustrated in Figures 3.3 and 3.4. In these figures we look at two models (logistic regression and LSTM), and we observe how the performance of the model (F1 score) changes as the number of posts from Stanford MOOC dataset are added to the training dataset. Our aim is to identify what number of added posts gives us the best performance. We start by adding 1000 posts and increase this number in increments of 1000 posts until we reach 5000 added posts. From both of these figures, we observe that as the number of added posts increase, the performance of the model drops. Similarly, from both of these figures, we observe that the setting which corresponds to 1000 added posts gives us the best performance.

Thus, for the experiments in this section we will augment the original training set with 1000 posts from the Stanford MOOCposts dataset in order to evaluate the impact that such a technique has on the overall model performance.

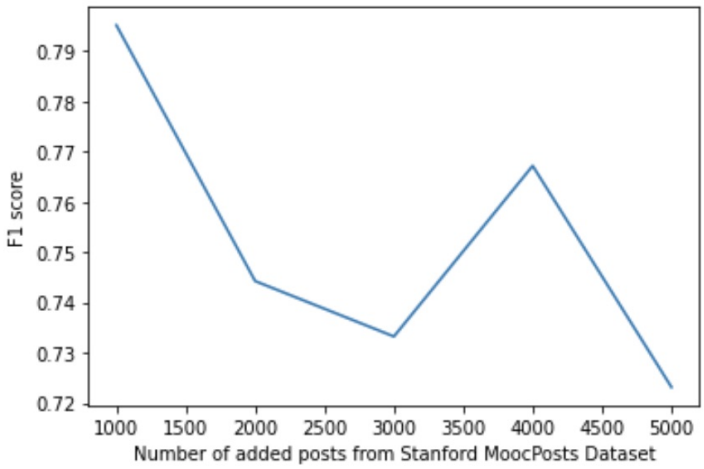

Figure 3.6: LSTM: change in F1 score performance based on number of Stanford MOOC posts added

Results

Since the Figures 3.5 and 3.6 show us that after 1000 posts, there is a strong decline in the model performance, we decided to carry out our experiments by adding 1000 examples from the Stanford MOOCPosts dataset.

Table 3.5 gives us information about the performance of the different models by training on an expanded dataset containing the CHEM dataset and 1000 examples from the Stanford MOOCPosts dataset. We can see that SVM gives the best performance of 85.3% F1 score on the validation set which is slightly less than the performance of SVM on the original CHEM dataset. Logistic regression and BERT are the second- and third-best models respectively. Again, SVM, LR and BERT make up the top-three models based on performance. However, as we observed with the Reddit data addition experiment, there was a decrease in the average F1 score of the models as compared to the original setting with no expanded training set.

Table 3.5: Performance of the models using dataset expansion from Stanford MOOCPosts dataset

Model	**Val. Acc.**	**F1 score**	**Precision**	**Recall**
Naive Bayes	0.843	0.760	0.756	0.803
Logistic Regression	0.920	0.848	0.914	0.799
Support Vector Machine	0.924	0.853	0.917	0.805
GRU	0.852	0.780	0.758	0.832
LSTM	0.873	0.798	0.788	0.818
BERT	0.911	0.846	0.893	0.816

However, one observation that we can make is that the RNN-based GRU and LSTM suffered sharp drops in performance: 6% drop in performance for both models. For comparision, BERT and SVM suffered around 1% and 0.8% drop in performance respectively. This shows that the RNN-based models are less robust to the mixed distribution (CHEM and Stanford MOOC) in the training dataset.

Another interesting take-away is the differences in the performances that are seen in Tables 3.4 and 3.5, i.e., comparison of model performance with the addition of Reddit data versus the addition of Stanford MOOCPosts data. In both instances we added 1000 new posts to the training set. We observe that all models except GRU and LSTM showed a better performance with the addition of the Stanford MOOCPosts data, whereas, the RNN-based models showed a better performance with the addition of Reddit Data. However, since majority of the models showed an improved performance with the Stanford data, we can draw the conclusion that the Stanford MOOCPosts dataset is closer to the distribution of the CHEM dataset.

In summary, we did not notice any performance improvements by augmenting with the Stanford MOOCPosts dataset.

3.7.3 Transfer Learning Using Stanford MOOCPosts Dataset

In this section we explore using Transfer Learning using Stanford MOOCPosts dataset. Transfer learning is an approach in which a model trained on a separate task or problem is adapted to solve another problem. A common technique for adapting a model is to fine-tune it on the new task. Fine-tuning refers to the technique of adapting the model on a new dataset by retraining it using the new data. Since transfer learning applies to deep learning models, we will only be performing experiments on GRU, LSTM and BERT. In these experiments we train these models first on the adapted Stanford MOOCPosts labeled dataset. Then, we fine-tune it to the CHEM data and then evaluate the validation performance on a held out semester's worth of data just like before.

Results

In Table 3.6, we display the results obtained from using transfer learning on models trained on the Stanford MOOCPosts dataset and fine-tuning it to the CHEM dataset. The primary purpose of trying the transfer learning experiment is to see if pretraining on a larger corpus of a similar domain could lead to performance improvements on the CHEM HS task. As we can see, transfer learning using BERT yielded the best F1 score on the validation set.

However, the F1 score of 84.9% is still less than the F1 score of 86% obtained by BERT on the original task. This shows that performance has dipped. However, when we examine the effects on the precision and recall, we see that while precision on the transfer learning model has dropped from 89.1% to 84.7%, the recall has increased from 83.3% to 85.6%. This shows that if the cost of not picking up a HS post is high, then transfer learning on BERT gives better performance as opposed to the original setting of only using CHEM data during training. This makes sense, because by pretraining on a large corpus of Stanford MOOCPosts data, the model has learned to recognize different kinds of ways in which HS behavior can be exhibited.

On the other hand, both GRU and LSTM have dropped in performance when compared to the original setting. This seems to align with a previous observation that both the RNN-based approaches are having trouble working with a mixed distribution dataset consisting of CHEM and Stanford MOOCPosts data.

3.8 Analysis of Results

In this section we experimented with different types of NLP models: classical ML, RNN and transformers and studied the HS classification performance in three different settings: using only CHEM data in training, using Reddit/Stanford data to expand training set and lastly, transfer learning by training on the entirety of the Stanford MOOCPosts dataset and fine-tuning on CHEM dataset. What we observed is that when computing the average validation performance on a semester's worth of CHEM data, we see best performance when we only use CHEM data in the training. Using data from other distributions does not improve validation performance on CHEM data.

Out of all the models and settings that we explored, we found that we obtained the best F1 score using the logistic regression classifier and using only CHEM data in the training set. Furthermore, we obtained the best precision score using the same logistic regression model and also in the same setting. Moreover, the best recall score was obtained by using a GRU trained on the CHEM data. Agrawal et al. [19] observed that logistic regression and SVM

Table 3.6: Performance of the models using transfer learning

Model	**Val. Acc.**	**F1 score**	**Precision**	**Recall**
GRU	0.891	0.807	0.800	0.823
LSTM	0.884	0.798	0.771	0.848
BERT	0.918	0.849	0.847	0.856

performed significantly better than naive bayes in the confusion text classification task that they worked on. Our work was able to corroborate these results, as we noticed that LR and SVM consistently outperformed naive bayes on our HS classification task.

When expanding our dataset using Reddit and Stanford data we found that all of the models suffered a drop in performance as compared to the original setting. When comparing between adding Reddit versus Stanford data to the training corpus, we observed that for the majority of models except the RNN-based models, adding Stanford data gave better performance than adding Reddit data. Furthermore, the RNN-based models suffered a large (6%) performance drop when trained on CHEM along with Stanford MOOCPosts data as compared to the original setting. This shows that the RNN models were less robust to the addition of Stanford MOOCPosts data as compared to the other models.

Transfer learning (using GRU, LSTM and BERT) on the other hand, proved to be a more effective approach than simply adding to the training dataset. We found that BERT gave the best result when using the transfer learning approach. Although the overall F1 score and precision of BERT dropped compared to the original setting, the recall however improved when compared to the original setting. This means that by pretraining on a larger corpus BERT identified different ways of expressing HS behavior, which in turn helped boost the recall. This result corroborates the findings made by Brahman et al. [24], wherein a multi-task learning setup lead to an increase in recall.

In conclusion, what we observed by experimenting using these different models in different settings is that in all of the different scenarios, LR, SVM and BERT gave the best and most robust performance. These were the top-three performing models in each setting (LR and SVM were not used in transfer learning setting).

3.9 Summary of Results

In this section we explored and identified the NLP models that ended up giving the best performance in the HS text classification task. Although logistic regression gave the best overall performance, we observed that SVM and BERT also ended up giving very similar performance but only slightly lower F1 scores.

Our initial hypothesis before diving deeper into the exploration was that expanding the CHEM dataset using data from similar domains as well as transfer learning could potentially boost performance and improve generalizability. We had identified these additional data sources as arising from a similar distribution as the CHEM dataset. However, upon further exploration, we found that expanding our original dataset and transfer learning both did not

yield any observable performance improvements in the text classification task on the CHEM data.

Chapter 4

SOCIAL NETWORK ANALYSIS OF CHEM DISCUSSION FORUM DATA

4.1 Introduction

In this chapter, we explore the interaction graphs in the CHEM discussion forum data, and determining to what extent, the amount of interaction in course discussion forums informs the course outcome (final course grades).

We will be constructing interaction graphs from the CHEM data and using these constructed graphs in our exploration study. Although we will be diving deeper into the construction of these graphs in a later section, the nodes of the graphs will represent students and an edge would indicate a reply between the nodes.

More specifically from a social network analysis perspective, we plan to study the relationship between:

1. The number of posts made by a student and their course outcome (grade).
2. The number of help-seeking posts made by a student and their course outcome (grade).
3. The number of replies to help-seeking posts made by a student and their course outcome (grade).
4. centrality/importance of a student in the discussion forum interaction graph and their course outcome (grade).

The critical question that we plan to answer in this chapter is if participating in course discussion forums is leading to a better student performance in the course.

4.2 Related Work

Williams-Dobosz et al. [27] study the impact that discussion forum engagement has to the course outcome in students traditionally underrepresented in STEM versus those students that do not belong in this category. This thesis adopts a similar undirected graph construction as was used in this paper, but a key distinction is that our work expands the study of centrality to directed graphs as well. Our work also uses a similar set of engagement metrics

that are outlined in this paper such as number of posts, number of HS posts made and replied to and also node centrality. One of the observations made in [27] is that help-seeking behavior was a significant contributor to course improvement.

Hecking et al. [30] explore the modelling of social and semantic roles of students interacting in course discussion forums.

Furthermore, Abnar et al. [31] look at extracting information about social roles within a network of social interactions. The paper proposes a new metric of using betweenness centrality along with other metrics to identify social roles in a network. We will also be using centrality in our work to identify students that have the most influence in the course discussion interaction graph.

4.3 Data

In this section, we will be outlining the characteristics of the dataset that we will be using for experiments in this chapter. We will only be using data from the CHEM discussion forums in this study. The difference between the CHEM data in this chapter, versus the CHEM data in the previous chapter, is that here, we will also be considering the replies ("Level-1") to the "Level-0" posts. However, we will not be looking at the replies to the "Level-1" replies, primarily because we do not have the help-seeking labels for "non Level-0" posts (replies) in this dataset. Thus, we will be constructing the interaction graph by taking into account "Level-0" posts and the replies to "Level-0" posts.

In the Tables 4.1, 4.2 and 4.3, we outline some of the important characteristics of the dataset across the seven semesters, which we will make use of in the experiments.

Table 4.1 contains a breakdown of the number of students that participated in the CHEM discussion forum across the seven semesters from which the data has been collected. An important point to note with regard to Table 4.1, is that it does not necessarily reflect the exact number of students that were enrolled in the semesters, because it is possible that some students who were enrolled in the semester did not participate in the discussion forum at all. All of the data that was collected, is anonymized and protects the privacy of the students as per FERPA (Family Educational Rights and Privacy Act) requirements.

Table 4.2 contains a breakdown of the number of posts and the number of HS posts across the seven semesters. One important note with regard to Table 4.2 is that number of posts in this table is slightly different when compared to the number of posts in Table 3.1 from Chapter 3. This is because the posts made by instructors or other course staff is not included in the experiments in this chapter, since we are building interaction graphs only including

the students. Table 4.3 on the other hand contains a breakdown of the number of replies and the number of replies to HS posts.

4.4 Building Graphical Representation of CHEM Discussion Forum Data

We build a graphical representation of the discussion forum data, in which the nodes represent students and an edge between two nodes represents that there was a reply between the two students. In the following paragraphs, we will go into more depth regarding the notion of the direction of edges.

One of the main reasons for constructing a graphical representation is that it gives us the information about centrality or importance of a student in the discussion forum interactions. The centrality can encode both help-giving and help-seeking centrality, depending on how the edges of the graph are directed.

Furthermore, in constructing the interaction graphs, we omit looping edges. This means that if a student replied to their own post, then we do not include that edge in our graphical representation. The reason we do so, is that these type of posts are usually clarifications made as a follow-up to the post, thus, they do not encode any information about discussions or interactions between one student and another.

Figures 4.1, 4.2 and 4.3 are the interaction graphs constructed from Semester 3 of the CHEM course.

Table 4.1: Number of students participating in the CHEM discussion forums across the different semesters

Semester	**Number of students**
Semester 1	60
Semester 2	30
Semester 3	79
Semester 4	11
Semester 5	14
Semester 6	57
Semester 7	25

Table 4.2: Breakdown of the number of posts and HS posts across the different semesters

Semester	**Number of posts**	**Number of HS posts**
Semester 1	297	271
Semester 2	362	59
Semester 3	1095	312
Semester 4	28	27
Semester 5	471	28
Semester 6	327	119
Semester 7	40	24

Table 4.3: Breakdown of the number of replies and replies to HS posts across the different semesters

Semester	**Number of replies**	**Number of replies to HS posts**
Semester 1	209	198
Semester 2	49	20
Semester 3	211	145
Semester 4	13	13
Semester 5	11	6
Semester 6	82	55
Semester 7	15	12

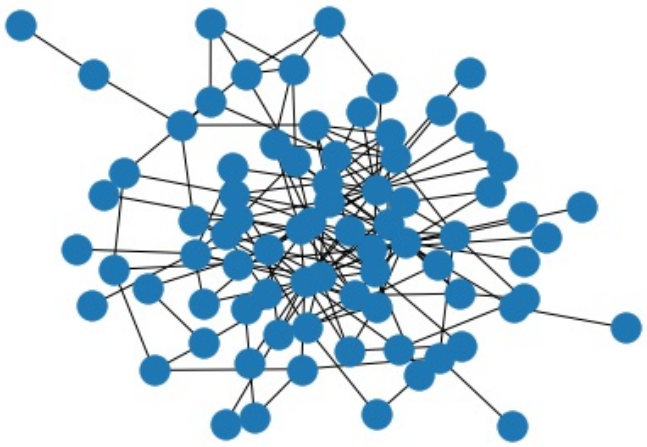

Figure 4.1: Undirected graph construction of student interactions in Semester 3

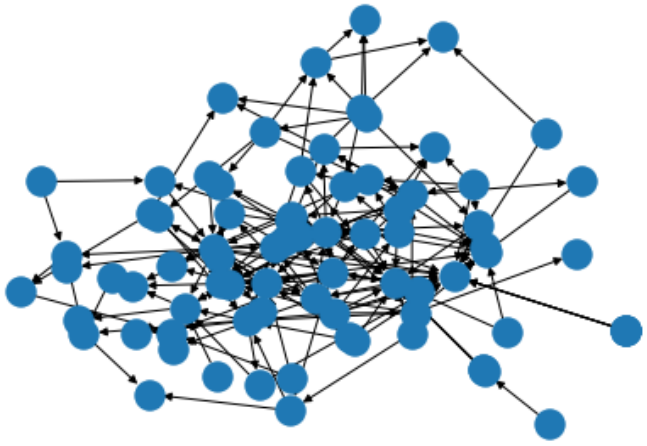

Figure 4.2: Type I directed graph construction of student interactions in Semester 3

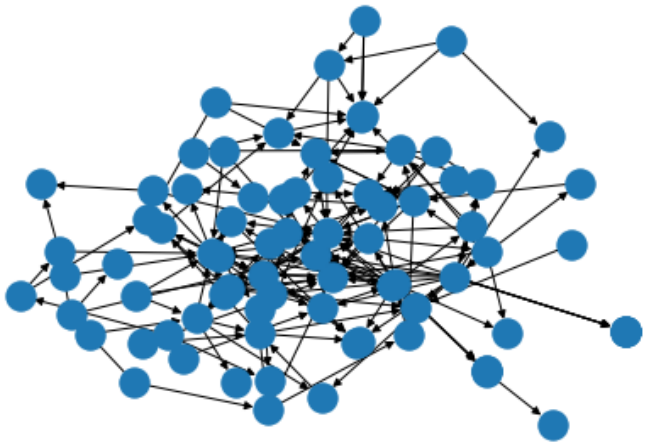

Figure 4.3: Type II directed graph construction of student interactions in Semester 3

Undirected Graph

The undirected graph representation does not encode any knowledge about the direction of the edges or replies. This means that there is an edge between two nodes if there exists a reply between the two students. It does not matter which one of the students made a post and which one replied. We will use the undirected graph representation as a basic version of the graph to measure the centrality of students and will serve as a baseline for more nuanced versions of the interaction graphs that we will proceed to explain below.

Type I Directed Graph

In the Type I directed graph, the edge between two nodes is directed toward the node (student) that received help.

Type II Directed Graph

In the Type II directed graph, the edge between two nodes is directed toward the node (student) that gave help.

Centrality Metrics

For both types of directed graphs, we will be using eigenvector centrality and pagerank centrality as the centrality metrics. Since pagerank centrality is only defined for directed graphs, we will use only degree centrality and eigenvector centrality as centrality metrics for the undirected graph respresentation. As a note, since the overall degree is the same for directed and undirected graphs, we will compute degree centrality only on the undirected graph construction.

4.5 Experiments and Results

In this section, we explore the relationship or correlations between the the different variables that we identified at the start of this chapter (Section 4.1) and the course outcomes. We compute the correlation of the two variables in consideration using Pearson's correlation coefficient. We also tabulate the p-values of the two variables. While the correlation tells us the linear correlation between the two quantities, the p-value tells us the probability that the correlation occurred by chance [32]. The value of the correlation coefficient ranges between

-1 and 1, where a positive value indicates a positive linear relationship and a negative value indicates a negative (inverse) linear relationship [32]. On the other hand, since a p-value is a probabilistic value, it ranges between 0 and 1. Furthermore, since we have seven semesters worth of data and since each semester contains different number of interactions (posts and replies), we will consider each semester separately, instead of computing the correlation and p-values as an aggregate over all semesters. We believe that this technique of separating the data from different semesters is a fair way of determining the correlations and would give us a better account of the data, because of the difference in the level of engagement across the semesters. Course outcomes or grades are available for each student that enrolled in and completed the course. Course grades have the values 1, 2, 3, and 4, where 4 implies the highest possible grade level and 1 implies the lowest possible grade level.

4.5.1 Correlation Between the Number of Posts Made by a Student and Their Course Outcome

As we can see in Table 4.4, in six out of the seven semesters, there is a positive correlation between the number of posts made by a student and their course outcome. The one semester in which there was a very small negative correlation (-0.062), there was a large p-value of 0.607, which means that there is a roughly 61% probability that the correlation occurred by chance. In the semesters that had the lowest p-values of 0.0 (Semester 2, Semester 3 and Semester 5), there was a relatively high positive correlation (0.427, 0.450 and 0.430 respectively) between the number of posts made and the outcomes achieved. This means that we found that students who were making more posts typically ended up receiving higher grades.

Table 4.4: Correlation between the number of posts made by student and their course outcome

Semester	**Correlation**	**p-value**
Semester 1	0.155	0.147
Semester 2	0.427	0.0
Semester 3	0.450	0.0
Semester 4	0.184	0.200
Semester 5	0.430	0.0
Semester 6	-0.062	0.607
Semester 7	0.067	0.510

4.5.2 Correlation between the Number of HS Posts Made by a Student and Their Course Outcome

As we can see in Table 4.5, in six out of the seven semesters, there is a positive correlation between the number of HS posts made by a student and their course outcome. The semester that had the lowest p-value of 0.043 (Semester 3), also had the highest correlation coefficient of 0.209 between the number of HS posts made and the outcomes achieved. However, the correlation coefficients are not as high as the ones observed in Table 4.4. By observing the trends in this table, we found that students who were making more HS posts typically ended up receiving higher grades.

4.5.3 Correlation between the Number of HS Posts Replied to by a Student and Their Course Outcome

As we can see in Table 4.6, in all of the seven semesters, there is a positive correlation between the number of HS posts replied to by a student and their course outcome. The semester that had the lowest p-value of 0.001 (Semester 2), also had the highest correlation coefficient of 0.396 between the number of HS posts replied to and the outcomes achieved. By observing the trends in this table, we found that students who were replying to more HS posts typically ended up receiving higher grades.

Table 4.5: Correlation between the number of HS posts made by student and their course outcome

Semester	**Correlation**	**p-value**
Semester 1	0.172	0.107
Semester 2	0.202	0.101
Semester 3	0.209	0.043
Semester 4	0.201	0.161
Semester 5	0.112	0.290
Semester 6	-0.096	0.427
Semester 7	0.070	0.491

4.5.4 Correlation between the Centrality of a Student in the Interaction Graph and Their Course Outcome

Undirected Graph Representation

We can see from Table 4.7 that for both, the degree centrality and eigenvector centrality, there was a positive correlation between the centrality of a student in the interaction graph and their course outcome in six of the seven semesters. Since centrality of a node in an interaction graph indicates importance or influence in the interactions, we can extrapolate that the positive correlation in a majority of the semesters indicates that typically, students that had a greater influence in the discussion forums ended up with higher grades than students who did not have as much influence.

Type I Directed Graph Representation

As a summary, in the Type I directed graph construction, the edges are directed toward the student that received help. This means that a higher centrality would typically indicate a higher involvement in the replies received. As we can see from Table 4.8, there was a positive correlation between pagerank centrality and course outcome in six out of seven semesters. Whereas, there was a positive correlation between eigenvector centrality of a student and their course outcome in all of the seven semesters.

Type II Directed Graph Representation

As a summary, in the Type II directed graph construction, the edges are directed towards the student that gave help. This means that a higher centrality would typically indicate a higher involvement in the replies given. As we can see from Table 4.9, there was a positive

Table 4.6: Correlation between the number of HS posts replied to by student and their course outcome

Semester	**Correlation**	**p-value**
Semester 1	0.284	0.007
Semester 2	0.396	0.001
Semester 3	0.221	0.032
Semester 4	0.096	0.508
Semester 5	0.092	0.384
Semester 6	0.202	0.091
Semester 7	0.133	0.193

Table 4.7: Correlation between the degree/eigenvector centrality of a student and their course outcome in an undirected graph construction

	Degree Centrality	**Eigenvector Centrality**
Semester	**Correlation (p-value)**	**Correlation (p-value)**
Semester 1	0.326 (0.002)	0.327 (0.002)
Semester 2	0.373 (0.002)	0.307 (0.012)
Semester 3	0.355 (0.0)	0.364 (0.0)
Semester 4	0.135 (0.350)	0.073 (0.615)
Semester 5	-0.044 (0.674)	-0.062 (0.555)
Semester 6	0.161 (0.179)	0.175 (0.145)
Semester 7	0.092 (0.368)	-0.002 (0.981)

Table 4.8: Correlation between the eigenvector/pagerank centrality of a student and their course outcome in an Type I directed graph construction

	Eigenvector Centrality	**Pagerank Centrality**
Semester	**Correlation (p-value)**	**Correlation (p-value)**
Semester 1	0.286 (0.007)	0.310 (0.003)
Semester 2	0.215 (0.080)	0.353 (0.003)
Semester 3	0.211 (0.041)	0.215 (0.038)
Semester 4	0.101 (0.484)	0.206 (0.152)
Semester 5	0.066 (0.531)	-0.003 (0.975)
Semester 6	0.245 (0.039)	0.262 (0.027)
Semester 7	0.053 (0.604)	0.104 (0.308)

correlation between pagerank centrality and course outcome in six out of seven semesters. Whereas, there was a positive correlation between eigenvector centrality of a student and their course outcome in five of the seven semesters.

4.6 Analysis of Results

When examining the correlation between the number of posts a student makes and their course outcome, we observe that in all of the semesters except Semester 6, there is a positive correlation between the two variables. When we consider p-values, we observe three semesters where the p-values were very close to 0.0 that had correlation scores of 0.427, 0.450 and 0.430. These also ended up being the highest correlation scores that we observed, thus we can say that these results are statistically significant. Furthermore the only negative valued correlation score was a very small negative value of -0.062. The associated p-value was a relatively high 0.607, making the correlation score not statistically significant. Thus, we can conclude that overall, there seems to be a positive correlation between the number of posts made and the course outcome.

When looking at the correlation between the number of HS posts a student makes and their course outcome, we again observe that in all of the semesters except Semester 6, there is a positive correlation between the two variables. The lowest p-value we observed was 0.043 for Semester 3 which also had the highest correlation score of 0.209, thus making the negative correlation statistically significant. Furthermore the only negative valued correlation score was a very small negative value of -0.096. The associated p-value was a relatively high 0.491, making the correlation score not statistically significant. An observation is that the correlation scores on average are not as high as we observed for the relation between number of posts and course outcome. Thus, we can conclude that there is a positive correlation

Table 4.9: Correlation between the eigenvector/pagerank centrality of a student and their course outcome in an Type II directed graph construction

	Eigenvector Centrality	**Pagerank Centrality**
Semester	**Correlation (p-value)**	**Correlation (p-value)**
Semester 1	0.233 (0.028)	0.239 (0.024)
Semester 2	0.098 (0.430)	0.242 (0.048)
Semester 3	0.316 (0.002)	0.331 (0.001)
Semester 4	-0.043 (0.769)	0.118 (0.413)
Semester 5	-0.148 (0.158)	-0.079 (0.456)
Semester 6	0.046 (0.706)	0.085 (0.481)
Semester 7	0.054 (0.597)	0.076 (0.455)

between the number of HS posts and the course outcome, but the positive correlation is not as strong as in the previous case when we considered the number of posts versus course outcome.

When examining the correlation between the number of HS posts replied to and the course outcome, we observe that there is a positive correlation between the two variables in all of the seven semesters. The semesters having the highest correlation scores also have the lowest p-values, making the observed relationship statistically significant.

Finally, when considering the centrality of a node in the interaction graph we considered three different centrality computations: degree centrality, eigenvector centrality and pagerank centrality. We computed these three centrality metrics for three different graph constructions: undirected, Type I directed and Type II directed. In all of these six different settings, a majority of the semesters (minimum majority we observed was 5 out of 7) exhibited a positive correlation between student centrality in the graph and the course outcome. In all of the six different settings, the highest correlation score was associated with the lowest p-value, making the positive correlation statistically significant.

4.7 Summary of Results

In this study, we examined the impact that interaction on course discussion forums have on the final outcome of the performance of a student. We looked at this problem from a social network perspective and tried to examine the correlation between engagement metrics (number of posts and number of replies) to course outcome and also centrality metrics of students in the interaction graph to course outcomes. We found that in an overwhelmingly large number of semesters, there was a positive correlation between these metrics to the grade that was received. While [27] found that help-seeking was a significant contributor to course improvement, our study validates and supplements this finding by demonstrating that help-seeking and help-giving are significant contributors to course performance. In Chapter 5 we explore the design of a UI/UX for course discussion forums with the aim of improving the help-giving and help-seeking experience for students on course discussion forums.

Chapter 5

UI/UX PROPOSAL FOR COURSE DISCUSSION FORUMS

5.1 Introduction

In this chapter, we explore the design of a discussion forum user interface/user experience based on what we have discussed and explored so far in this thesis. An important feature of the discussion forum experience that we propose is integrating automated help-seeking classification in the discussion forum interface. We have already explored the task of classifying posts as help-seeking or not help-seeking in Chapter 3 of the thesis. We can directly apply the models developed into this feature for the discussion forum.

We can also apply the computation of centrality and other interaction metrics that were explored in Chapter 4 into a "participation" score for students.

We will make use of the principles outlined in Google's People + AI Guidebook [33] to inform design choices in this chapter. This resource is meant to serve as a guide for developers and designers building human-centered ML systems.

5.2 System/Interface Design

The interface mock-ups included in this section were prepared using Figma, where we made use of the Figma Wireframing Kit. We also used a Lorem Ipsum generator (`https://loremipsum.io/`) to generate random text that made up the post content in the mock-ups. Figures 5.1 and 5.2 give a general overview of two discussion forum views. One can filter the discussion posts by help-seeking only posts or choose to browse the unfiltered view. These options are included in the sidebar. Furthermore as can be seen, help-seeking posts are tagged using a green badge.

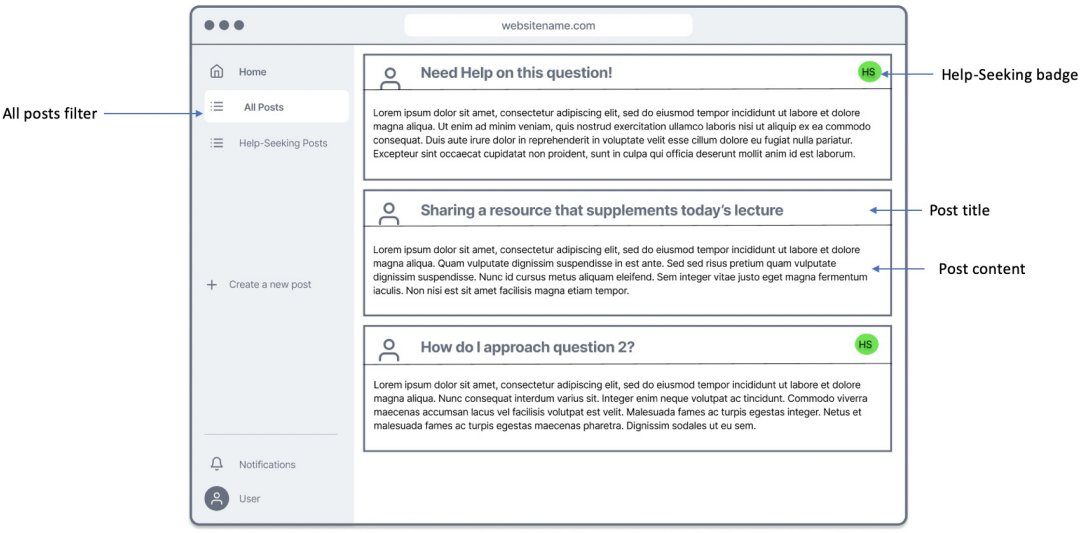

Figure 5.1: Discussion forum "All Posts" view

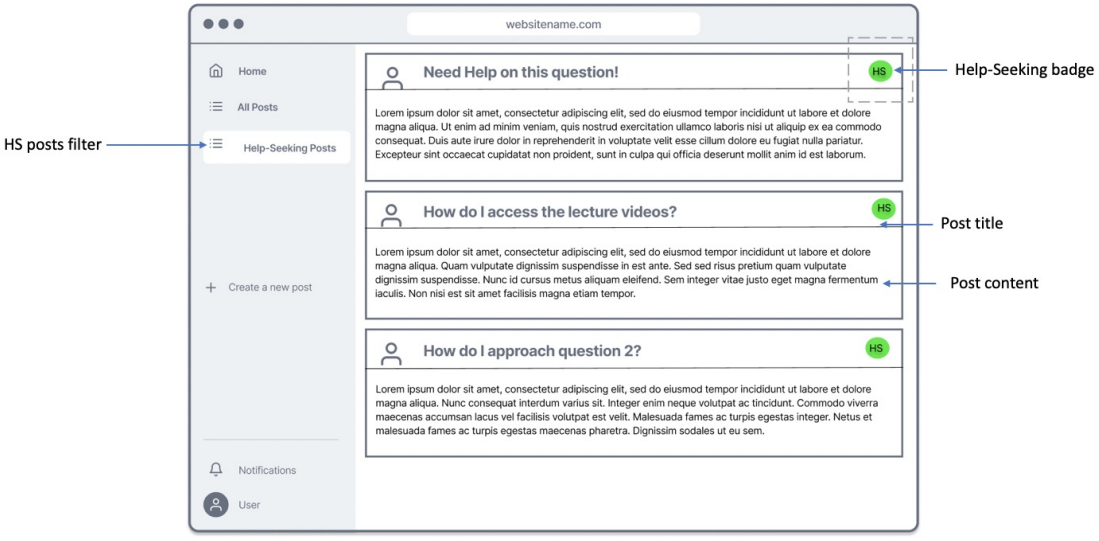

Figure 5.2: Discussion forum "Help-Seeking Posts" view

5.2.1 User Needs

Before we go into any further discussion, it is important to clearly outline the user needs and what problem we plan to solve with our UI/UX for course discussion forums. Here is a list of the specific user needs that we believe our system should be trying to address:
1. To try to reduce the time it takes for a student to get a response to a help-seeking question.
2. To try to make it easier for course staff and students to identify help-seeking questions so that they can be attended to quickly and efficiently.
3. Make it easier for students to understand their participation effort in discussion forums and also for instructors to understand and evaluate the general as well as specific interaction patterns in the forum. This information from an instructor's point of view could also be used for participation grade determination if a course does have such a requirement.

The Figures 5.1 and 5.2 from the previous section address user need number 2. The ability to filter posts by the help-seeking label aims to achieve this objective.

5.2.2 Data Collection

When we are dealing with student interaction activity and the associated posts and replies made by students, we need to ensure that this data that is securely stored and appropriately anonymized. If discussion posts are collected for any purpose (e.g. augmenting the text corpus of discussion posts for training the HS classifier), then this must be an opt-in feature for students. Facebook's Responsible Innovation Principles [34] state that "Never surprise people", and this principle should be applied to the data collection feature of our discussion forum user experience. It must be up to a student to opt-in to such a feature. Even after a student opts in to the feature, appropriate care must be taken to anonymize any student identifying information from the text data.

5.2.3 Notification Driven Help-Giving

We want to encourage students to give assistance to their peers who are in need of help. For this reason, we have identified that prodding students (via a notification) to help out someone who needs help is a quick and an efficient manner to encourage help-giving behavior. Figure 5.3 demonstrates how such notifications can be designed. In order for a feature to be effective it must aim to reduce friction to achieve a given task. The notification design in Figure 5.3 provides a direct link to a help-seeking post that needs attention. In order

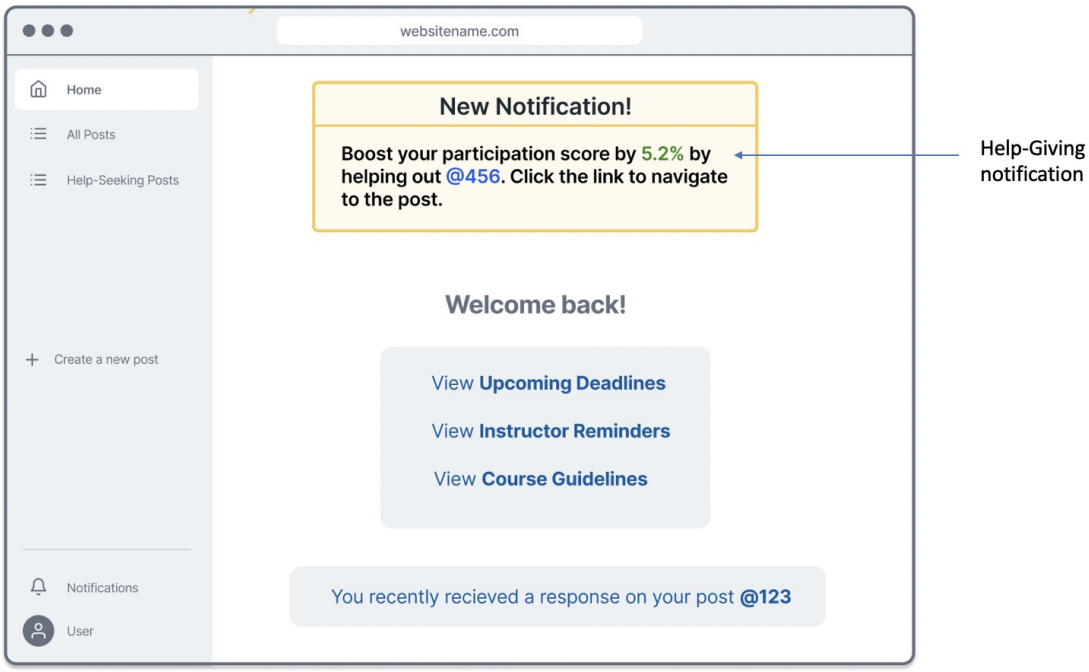

Figure 5.3: Notification driven help-giving

to determine which help-seeking posts should be prodded to users, we can implement a functionality in the backend to sort unanswered help-seeking posts based on the amount of time that they have remained unanswered. The posts that have remained unanswered for the longest amount of time can then be directed to students via notifications. The "Feedback + Control" chapter of Google's PAIR Guidebook states that it is important to explicitly communicate the value of impact to users in order to build on their existing mental models of the system. We incorporated this principle to our notification design by communicating the impact or value that the help-giving behavior would have to their own participation score. This not only encourages help-giving behavior but also explicitly communicates the impact that the help-giving behavior would have.

5.2.4 Automated HS Classification while Creating a Post

This feature automatically classifies student posts as HS or not when a student creates a new post. In order to implement this feature, we need to ensure that the recall of our classifier is high. This means that our classifier should be able to capture a high fraction of all of the HS posts. This could mean that our classifier wrongly classifies some posts that are not HS, but that is acceptable because the cost of predicting a HS post as non-HS is higher

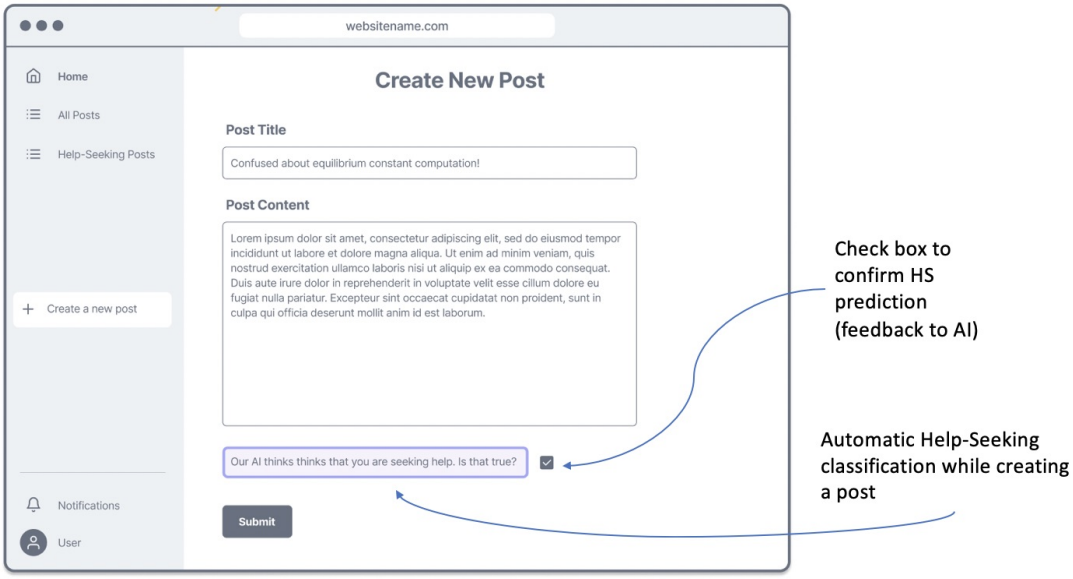

Figure 5.4: Automated HS classification while creating a post

than the cost of predicting a non-HS post as HS. Furthermore, automatically predicting HS behavior reduces the friction that students face while creating a post and reduces the likelihood of students intentionally classifying a non-HS post as HS, simply to get a faster reply. Furthermore, we have the option for students to confirm the prediction of our HS classifier, thus allowing for graceful failure and control, which are two principles highlighted in the People + AI guidebook. This feature is demonstrated in Figure 5.4.

5.2.5 Forum Interaction Insights

When evaluating the interaction performance of a student in a discussion forum (whether for grading or otherwise), it would be helpful to provide the relevant breakdown as illustrated in Figure 5.5. Not only is it important to know how many posts or replies a student made, but it is also useful to gain insights about the centrality of a student with respect to the discussions as well as the help-seeking posts or replies to help-seeking posts that a student makes. This insights give a more complete picture of the interactions of a student in the discussion forum and it is important not only from a student's perspective but also from a participation grade assignment perspective, if a course has such a requirement. Even in the absence of participation grading, these insights can be used to assign badges to students that actively participate or help out other students, thus encouraging participation. Course

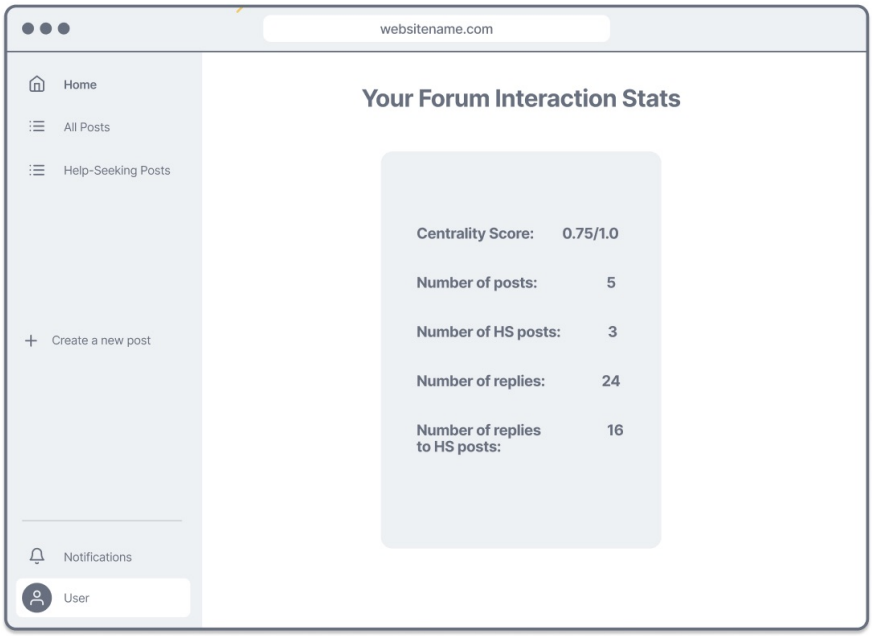

Figure 5.5: Forum interaction insights

discussion forums like Piazza and Campuswire assign such badges for students that meet a specified threshold of posts or upvotes received.

Chapter 6

CONCLUSION

In the Chapter 3, we explored the help-seeking text classification using classical ML models (NB, LR and SVM), RNN-based models (GRU and LSTM) and a transformer-based approach (BERT). We found success using logistic regression, support vector machine and BERT in the text classification task. We also identified two similar domain data sources that included r/HomeworkHelp subreddit and Stanford MOOCPosts dataset to augment the training corpus that consisted of CHEM posts. We did not find this technique of corpus expansion to directly translate to performance improvements in the text classification task. We then experimented using transfer learning on the Stanford MOOCPosts dataset and fine-tuning on the CHEM dataset. This approach was applied on the GRU, LSTM and BERT models. While it did not lead to a better performance than the best performing model (logistic regression) in the original setup, it did indicate that transfer learning via BERT improved the recall score, when compared to the original BERT model. From a bigger picture perspective, we found that using logistic regression by training only on the CHEM data gives us the best F1 score of 86.5%.

We then examine the interaction patterns in the CHEM dataset and we try to determine if an increased level of discussion forum activity correlates to a better course outcome. From our experiments we found a generally positive correlation between activity in the CHEM discussion forums and course outcomes.

Equipped with this knowledge, we propose a UI/UX for course discussion forums that makes it easier to get help and give help on course discussion forums. Improving the wholistic product experience for the end users is critical to making the course discussion forums a useful part of MOOCs and other online learning experiences.

Future Work

One of the future extensions of the work in this thesis would be to gather data from different course discussion forums and evaluate if the text classification models developed in this work generalize well to help-seeking text classification. One challenge associated with this is to

find good quality labeled data in line with the HS categorization that was shown in this work.

Another potential extension includes evaluation of the kind of correlation that exists between student activity and course outcome in data from other course discussion forums and MOOCs. This would help us understand this problem statement from a broader perspective.

Lastly, the other possible extension to this work is to implement and deploy (in a real-world setting) a course discussion UI/UX in line with the principles and characteristics that we discuss in Chapter 5 and evaluate to what extent this redesign leads to a change in student behavior and course outcome.

BIBLIOGRAPHY

[1] M. Kichu and M. Bhattacharya, "Covid-19 pandemic impels surge in MOOC learning and the new normal: A literature review," *International Journal of Innovative Research in Technology*, vol. 7, no. 10, pp. 282–285, 2021.

[2] M. N. AlJeraisy, H. Mohammad, A. Fayyoumi, and W. Alrashideh, "Web 2.0 in education: The impact of discussion board on student performance and satisfaction." *Turkish Online Journal of Educational Technology-TOJET*, vol. 14, no. 2, pp. 247–258, 2015.

[3] D. Jurafsky and J. H. Martin, "Speech and language processing. vol. 3," *US: Prentice Hall*, 2014.

[4] A. Ng, "CS229 lecture notes," *CS229 Lecture notes*, vol. 1, no. 1, pp. 1–3, 2000.

[5] D. Forsyth, *Applied Machine Learning*. Springer, 2019.

[6] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, 2016.

[7] J. Eisenstein, "Natural language processing," 2018.

[8] C. Olah, "Understanding LSTM networks," 2015. [Online]. Available: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

[9] S. Gao, Y. Huang, S. Zhang, J. Han, G. Wang, M. Zhang, and Q. Lin, "Short-term runoff prediction with GRU and LSTM networks without requiring time step optimization during sample generation," *Journal of Hydrology*, vol. 589, p. 125188, 2020.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[11] S. Ruder, M. E. Peters, S. Swayamdipta, and T. Wolf, "Transfer learning in natural language processing," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, 2019, pp. 15–18.

[12] W. M. Kouw and M. Loog, "An introduction to domain adaptation and transfer learning," *arXiv preprint arXiv:1812.11806*, 2018.

[13] G. Strang, *Linear Algebra and its Applications.* Belmont, CA: Thomson, Brooks/Cole, 2006.

[14] M. M. Fleck, *Building Blocks for Theoretical Computer Science (Version 1.3b)*, 2017.

[15] J. Zhang and Y. Luo, "Degree centrality, betweenness centrality, and closeness centrality in social network," in *Proceedings of the 2017 2nd International Conference on Modelling, Simulation and Applied Mathematics (MSAM2017)*, vol. 132, 2017, pp. 300–303.

[16] B. Ruhnau, "Eigenvector-centrality—a node-centrality?" *Social Networks*, vol. 22, no. 4, pp. 357–365, 2000.

[17] K. Henni, N. Mezghani, and C. Gouin-Vallerand, "Unsupervised graph-based feature selection via subspace and pagerank centrality," *Expert Systems with Applications*, vol. 114, pp. 46–53, 2018.

[18] A. Agrawal and A. Paepcke, "The Stanford MOOCPosts data set." [Online]. Available: https://datastage.stanford.edu/StanfordMoocPosts/

[19] A. Agrawal and S. Leonard, "Mining for confusion: Classifying affect in MOOC learners' discussion forum posts."

[20] S. A. Geller, K. Gal, A. Segal, K. Sripathi, H. G. Kim, M. T. Facciotti, M. Igo, N. Hoernle, and D. Karger, "New methods for confusion detection in course forums: Student, teacher, and machine," *IEEE Transactions on Learning Technologies*, vol. 14, no. 5, pp. 665–679, 2021.

[21] S. A. Geller, N. Hoernle, K. Gal, A. Segal, A. X. Zhang, D. Karger, M. T. Facciotti, and M. Igo, "# confused and beyond: detecting confusion in course forums using students' hashtags," in *Proceedings of the Tenth International Conference on Learning Analytics & Knowledge*, 2020, pp. 589–594.

[22] Z. Zeng, S. Chaturvedi, and S. Bhat, "Learner affect through the looking glass: Characterization and detection of confusion in online courses." *International Educational Data Mining Society*, 2017.

[23] A. Bakharia, "Towards cross-domain MOOC forum post classification," in *Proceedings of the Third (2016) ACM Conference on Learning@ Scale*, 2016, pp. 253–256.

[24] F. Brahman, N. Varghese, S. Bhat, and S. Chaturvedi, "Effective forum curation via multi-task learning," in *EDM*, 2020.

[25] Z. Zeng, S. Chaturvedi, S. Bhat, and D. Roth, "DiAd: Domain adaptation for learning at scale," in *Proceedings of the 9th International Conference on Learning Analytics & Knowledge*, 2019, pp. 185–194.

[26] X. Wei, H. Lin, L. Yang, and Y. Yu, "A convolution-LSTM-based deep neural network for cross-domain MOOC forum post classification," *Information*, vol. 8, no. 3, p. 92, 2017.

[27] D. Williams-Dobosz, R. F. L. Azevedo, A. Jeng, V. Thakkar, S. Bhat, N. Bosch, and M. Perry, "A social network analysis of online engagement for college students traditionally underrepresented in STEM," in *LAK21: 11th International Learning Analytics and Knowledge Conference*, 2021, pp. 207–215.

[28] V. Jay, G. Henricks, C. Anderson, L. Angrave, N. Bosch, N. Shaik, D. Williams-Dobosz, S. Bhat, and M. Perry, "Online discussion forum help-seeking behaviors of students underrepresented in STEM," 2020.

[29] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[30] T. Hecking, I. A. Chounta, and H. U. Hoppe, "Role modelling in MOOC discussion forums," *Journal of Learning Analytics*, vol. 4, no. 1, pp. 85–116, 2017.

[31] A. Abnar, M. Takaffoli, R. Rabbany, and O. R. Zaïane, "SSRM: Structural social role mining for dynamic social networks," *Social Network Analysis and Mining*, vol. 5, no. 1, pp. 1–18, 2015.

[32] D. M. Diez, C. D. Barr, and M. Cetinkaya-Rundel, *OpenIntro Statistics*. OpenIntro Boston, MA, USA:, 2012.

[33] "People + AI Guidebook." [Online]. Available: https://pair.withgoogle.com/guidebook

[34] "Responsible Innovation Principles | Meta." [Online]. Available: https://about.facebook.com/realitylabs/responsible-innovation-principles/