# ABSTRACT

HAQUE, AMANUL. Prioritizing Blended Course Discussion Forum Posts using Linguistic Features and Metadata with Limited Labelled Instances. (Under the direction of Dr. Collin F. Lynch).

Discussion forum is an integral element common across all MOOCs (Massive Open Online Courses) and Blended courses that enables interaction between various participants and instructors. Ease of access and few prerequisites for participation has contributed to the widespread popularity of MOOCs, with increased diversity in courses offered and platforms that offer them. High student to instructor ratio can lead to the problems of information overload and chaos in discussion forums due to the high volume of posts each day. This is specifically challenging for instructors who are often overwhelmed with the high volume of posts many of which are not content-related or are non-urgent. Prioritizing discussion forum posts based on its urgency could greatly improve accessibility and navigation for instructors on these forums and enhance student's learning and reduce drop-outs.

In this work, I explore the possibility of designing models that can be trained on limited labelled instances to rank posts based on its urgency in discussion forums. To understand urgency with regards to these discussion forums, I surveyed twelve experienced Teaching Assistants (TAs) for blended courses in this study. While the survey showed little consensus among the TAs as to what makes a post urgent, the survey did reveal some useful findings, such as: major factors that make a post urgent/non-urgent as perceived by a TA. I propose a semi-supervised learning model that can be trained on limited labelled instances from the survey to categorize posts into urgent and non-urgent. I used linguistic features and few metadata for training these models. To further improve the performance of my model, I used supervised models trained on other datasets and partially-supervised models trained on one-class. The results demonstrate that the semi-supervised learning alone performs moderately but showed improvements when combined with other models.

I also explored cross-domain classification to identify question posts using datasets from MOOC as well as blended courses. Cross-domain classification usually performs poorly for a multi-class

classification across MOOCs and blended courses because of the difference in discussion content and the distribution of types of posts between them. My model achieved good results in identifying questions in a blended course discussion forum using models trained on MOOC datasets based on linguistic features and custom word features. This presents us with an opportunity to explore cross-domain classification models further to be able to harness similarities across different MOOCs and blended course discussion forums to train more sophisticated models.

# Prioritizing Blended Course Discussion Forum Posts using Linguistic Features and Metadata with Limited Labelled Instances

by Amanul Haque

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Computer Science

Raleigh, North Carolina

2019

APPROVED BY:

| | |
|---|---|
| _____ | _____ |
| Dr. Tiffany Barnes | Dr. Sarah S. Heckman |

_____
Dr. Collin F. Lynch
Chair of Advisory Committee

## DEDICATION

*To my beloved late cousin Abdul Aziz.*
*Your altruism and optimism will*
*always be an inspiration for me.*

# BIOGRAPHY

Amanul Haque was born in Varanasi, India in 1990. He obtained his Bachelor's in Information Science & Engineering from PES Institute of Technology, India, in 2015. He joined the CEI (Center for Educational Informatics) Lab during his master's starting in 2017 at North Carolina State University. His research interest spans across machine learning and natural language processing.

## ACKNOWLEDGEMENTS

I would like to thank my advisor Dr. Lynch for all his support and guidance throughout my work at CEI lab at NCSU. I would also like to extend my gratitude to my committee members, Dr. Heckman and Dr. Barnes who provided useful suggestions during the course of this research study.

A special thanks to all the teaching assistants who took out time to participate in the survey for this study and provided their valuable inputs. Adam Gaweda for his suggestion on improving the survey interface and content. Yiqiao and Niki for their support and help with data collections and datasets. And a big thank you to all the members of the CEI lab I had the pleasure of working and interacting with.

## TABLE OF CONTENTS

# LIST OF TABLES

## LIST OF FIGURES

CHAPTER

1

# INTRODUCTION

Over the past decade, technology has increasingly been used to enhance course and content offerings both in the traditional face-to-face and online education settings. Massive Open Online Courses (MOOCs) aim to provide easy access to high-quality online study resources to a global audience on a large scale [1]. Since their inception in 2008, MOOCs have become increasingly popular and widespread. From late 2011 to 2016, more than 58 million users enrolled in over 6850 courses offered by more than 700 institutions globally [2]. MOOCs are typically free to access and have few prerequisites for participation and as a consequence, have attracted hundreds of thousands of people to register in the past few years and the number of users has increased rapidly due to the growth of novel platforms and the wide range of offered courses [3] [4].

Online learning technologies are transforming traditional brick and mortar classrooms by increasingly leveraging the power of technology through blended learning. Blended Learning is an education approach that combines online education tools with traditional face-to-face classrooms to enhance student's learning. Student achievement is shown to be higher in blended learning experiences when compared to either entirely online or entirely face-to-face learning experiences [5]. An important online tool common across all MOOCs and blended courses is the discussion forum that enables interaction between various participants and instructors. Prior researchers have shown that discussion in classrooms can facilitate student learning on course

content and student participation in online asynchronous discussions can enhance their learning [6] [7]. Online discussions have been suggested to be less threatening than speaking in front of a group in a classroom setup, thus making online discussion more favorable for many students [8]. Discussion forums also serve as a useful resource for instructors to better understand the gaps in learner knowledge and offer targeted feedbacks [9].

However, large number of posts on discussion forums in an online environment can hamper effective communication and decrease the level of interactivity among learners [10]. With thousands of active participants, these discussion forums are often flooded with huge number of posts each day making it difficult for the instructors to review and respond to all the posts. This high volume of daily posts hinders both instructors and learners to effectively navigate through the discussion forum to identify posts that are relevant. Furthermore, the high ratio of learners to instructor makes it more challenging for the instructor to be selective in the intervention [11], considering that most posts on these discussion forums are not content related [12]. Instructors have emphasized the need for better navigation mechanism for MOOC discussion forums for effective learning [13]. Students often have to make active efforts to repeatedly log-in to the discussion forum and wait for an unknown and variable amount of time for responses to their posts [14]. Too much lag-time waiting for a reply violates student's expectations and need for acknowledgement, leading them to disengage resulting in reduction in discussion's momentum [15]. Prior research has also revealed correlation between participant confusion and dropouts [16] and lack of responsiveness in MOOC forums could be a contributing factor in student dropouts [17] [18].

Through this research study, I explore the possibility of designing machine learning models that can categorize discussion posts to facilitate better navigation. This study focuses on separating question posts from non-question posts and further tries to find urgency within the question posts. Identifying urgent posts can make navigation more effective for instructors who are often overwhelmed by the sheer volume of posts on these discussion forums each day, most of which are either non-urgent or unimportant. Better mechanisms for instructors to identify urgent posts can help reduce the number of urgent questions going unanswered which can positively impact student's learning and reduce drop-outs.

CHAPTER

2

# RELATED WORK

Discussion forum in MOOCs is a tool used by instructor to facilitate interaction among learners and between learners and instructors to improve the quality of learning [19]. A study examining MOOC discussion forums revealed that the average number of newly created posts in 3 MOOCs with 1146, 771 and 24,963 active participants are as high as 96, 152 and 510 posts per day, respectively [20]. In another larger 7-week MOOC with more than 50,000 enrollments reported more than 50,000 posts [21]. This high volume of daily posts hinders effective navigation through the discussion forum for both instructors and learners to identify posts that are relevant.

Several approaches have been proposed to categorize student discussion forum post to facilitate more effective navigation and study correlation with student's learning outcomes. Marzouk et al. [22] based their analysis on the ICAP framework [23] which focused on the value of different dialogue contributions and their relation to student's learning process. The ICAP (Interactive, Constructive, Active, Passive) framework developed by Chi et al. [23] has also been widely used in assessing student engagement in other venues. Vellukunnel et al. [24] used ICAP framework to analyze student engagement behavior on blended course discussion forums for students enrolled in CS2 courses provided by two universities. They found positive correlation between student engagement on discussion forums and their performance in terms of final grades. They concluded encouraging questions that reflect reasoning or that connect the student's attempts to solve the problem could yield better learning outcomes.

Wise et al. [12] proposed a model for automatic identification of content-related posts to address the problem of information overload and chaos in MOOC discussion forums. Their model used unigram and bigram linguistic features and achieved accuracy > 0.8, kappa > 0.52, recall > 0.72 and precision > 0.62. They also concluded that stop-words improved the performance of their topic-based classification model whereas metadata information such as, number of votes and views wasn't helpful. Wei et al. [25] applied LSTM based deep neural network for cross domain forum post classification. Their model used transfer learning for confusion, urgency and sentiment detection in MOOC posts.

Many researches on student discussion forums has focused on automated mechanisms to provide timely support to learners. Chandrasekaran et al. [26] used L1-regularized logistic regression to predict whether an instructor should intervene in a discussion thread or not, while Chaturvedi et al. [11] proposed predicting instructor intervention using lexicon features and thread features. Answers to questions on student discussion forums can be ranked automatically based on helpfulness to reduce the load for other users to find the correct or best answer [27]. To solve the problem of unresolved questions in student online discussion forums, Kim et al. [28] proposed a model to automatically identify unresolved questions. Their model captures structural dependencies within the threaded discussion to identify discourse acts. They used Support vector machines and Transformation-based Learning (TBL) to assign 'speech acts' to individual posts to discern its role in the discussion thread and achieved an F score of 0.76 in discussion thread classification. Yang et al. [29] uses structural equation modeling (SEM) techniques to identify the resolvability of threads in MOOCs. Given a thread with its associated question and set of replies they try to identify whether it has been resolved or not using latent variable modeling.

Almatrafi et al. [30] proposed a supervised model that uses linguistic features, metadata (e.g. number of reads, up-counts and post type) and term frequencies to categories posts into urgent and non-urgent. They achieved a weighted F1 score = 0.88, f1 >= 0.70 and Kappa >= 0.63, and concluded that linguistic features such as term-frequency and features extracted from LIWC (Linguistic Inquiry and word counts) along with few metadata can be effectively used to identify urgent posts [30]. Agrawal et al. [31] proposed an approach to automate providing help via video recommendation for posts that express confusion. They used classification approaches to identify

posts that express confusion, followed by a recommendation to the start time of a video clip of a tutorial lecture from the course that specifically addresses that confusion. They also concluded that confusion was significantly correlated with questions, answers, urgency, sentiment and opinion [31].

This study differs from previous studies on urgency of MOOC posts on two major counts. First, it distinguishes urgency from posts that express confusion or posts that have an explicit/implicit question. Earlier works on urgency have treated posts expressing confusion or asking a question as urgent. In this study I am trying to find urgency within question posts, i.e. two questions may have varying degree of urgency based on many factors. For instance, a question concerning issue in submission due to a server failure or a last-minute question before an assignment deadline, are more urgent than a curiosity question that a student might ask to enhance his understanding of the course content. Second, I am using a very limited labelled instances to train a semi-supervised model in combination with models trained on other MOOC courses to rank posts in blended course discussion forum based on its urgency. Also, in order to establish an understanding of urgency for this work from an instructor's view point, I surveyed twelve experienced TAs from blended course.

CHAPTER

3

# CURRENT STUDY

In this study, I try to address the overload problem in MOOC discussion forums by designing a model that can identify urgent posts based on lexical features and meta-data. Urgent posts can be defined as those posts that demands immediate response from the instructor. A post that don't demand a response doesn't qualify to be considered urgent. An effective monitoring of urgent posts can aid instructors in prioritizing their responses and better manage the large volume of posts on online discussion forums. Prioritizing MOOC posts can help free up instructor's time to engage in other constructive activities that can improve MOOC quality and learning outcome.

## 3.1 Research Questions

In order to achieve this goal, several research questions need to be answered.

*RQ1*. What is urgency with regards to student discussion posts? How do teaching assistants and instructors view posts and what factors do they take into account in deciding whether a post is urgent or non-urgent?

*RQ2*. Can textual features in the post and some metadata reliably identify urgent posts?

*RQ3*. Can we train a robust model on limited labelled instances and achieve good performance?

*RQ4*. Can classifiers trained on one MOOC course be effectively used on other MOOCs and blended courses?

CHAPTER

4

# METHODS

I propose a model to classify posts into urgent and non-urgent posts using combinations of supervised, partially-supervised and semi-supervised learning models. First, we annotated the MOOC datasets BDE2013 and BDE2015 using our own coding schema to train a supervised classifier, the coding scheme is discussed in more detail under section 4.2. The blended course dataset JAVA2015 was partially labelled (150 instances labelled) by multiple experienced TAs through a survey, the survey results are discussed in more detail under the section 4.4.

As a first approach I trained a semi-supervised model using the labelled 150 instances and unlabeled data from JAVA2015 and validated using stratified 10-fold cross-validation. To improve the results, I used a partially-supervised classification model trained on 150 labelled instances to identify questions and non-questions in the unlabeled dataset, before applying the semi-supervised model. I designed 2 variations of this approach, these models are discussed in more detail under the section 5.4.

For the second approach, I combined Semi-supervised learning model with supervised-learning models trained on combined BDE2015 and BDE2013 dataset. The supervised learning models were trained to identify whether a post is a question or a non-question in the unlabeled JAVA2015 dataset and the semi-supervised model then uses this information along with 150 labeled instances

to categorize posts into urgent and non-urgent. This approach is discussed in more detail under the section 5.4.

### 4.1 Datasets

Three datasets were used in this research study. Two MOOC datasets on 'Big data in Education' provided by The Teacher's College at Columbia University and hosted on Coursera (BDE2013) and EdX (BDE2015) platforms in 2013 and 2015 respectively and one blended course dataset (JAVA2015) provided by North Carolina State University offered by Dr. Heckman on 'Programming Concepts - Java' in 2015.

The 'Big data in Education' is an eight-week course that includes materials for a graduate-level course of educational data mining and big data analysis in education. The curriculum introduces the basic methods of how to collect and analyze data such as techniques of visualization and clustering. The students were required to learn how to use data analysis software that was provided by the instructor and to use publicly available libraries. They then applied those tools to answer numeric or multiple-choice mastery questions. In the 2013 class, 778 students made at least one post or comment on the discussion forum with a total of 603 discussion threads consisting of 4259 posts in total. In 2015, 519 students produced 625 discussion threads with a total of 2056 posts.

The 'Programming Concepts - Java' is a blended course offered on-campus that uses Piazza as the online discussion forum platform. The course typically consisted of concepts from Java as a programming language and most of the assignments were coding assignments. Students were expected to learn Java concepts through assignments and were required to submit a working code to a Jenkins server for each assignment. Multiple teaching assistants (TAs) were at student's disposal to help them in the course, a major part of their activity was to promptly respond to student queries on the discussion forum. The discussion forum witnessed participation by 144 students who posted at least one post in the discussion amounting to 1099 discussion threads and a total of 2659 posts.

## 4.2. Coding schema for BDE2015 and BDE2013 datasets

The BDE2013 and BDE2015 datasets were initially tagged using an optimized ICAP framework schema [23] based on an initial review of our dataset. In order to capture the student's non-engagement behaviors, we also added politeness expression, social connection, and off-topic categories into our scheme. The scheme contains 12 classes representing different types of cognitive engagement in the discussion. Three experienced educational data mining researchers annotated the BDE2013 and BDE2015 datasets using the coding schema described in section 4.2. Each of the BDE datasets were annotated by two researchers independently, with me annotating the BDE2015 dataset (interrater agreement is discussed in section 4.5). Table 1 shows the definition and examples for each category and table 2 shows the distribution of posts across various categories for the 2 datasets.

For the purpose of this study, i.e. prioritizing discussion posts based on urgency, I merged all these categories into two, making it a binary classification problem. I merged all non-question categories (O, S, P, B, C, M and A) into one category and all the question categories (Q, BQ, CQ) into another. I excluded all the other language posts and off-topic posts (category X, T) from this study as such posts have little to no occurrence in a blended course discussion forums. This divides the dataset into two labelled categories, first, posts that pose a question and hence demand a response, second, answers or statement posts that do not necessarily demand a response. This resulted into a distribution of {Question: 973, Non-question: 2645} with a total of 3618 posts for the BDE2013 dataset and {Question: 360, Non-question: 1488} with a total of 1848 posts for the BDE2015 dataset.

Table 1. Coding Schema for BDE datasets

| Category | Definition | Example |
|---|---|---|
| **On topic question (Q)** | Ask questions related to course topics to understand and complete their homework. Excludes posts that fit under BQ (software issue questions) or CQ (course relevant questions). | How does the teacher want the answer to be? percentages or decimal? |
| **On topic statement (O)** | Provide an answer or opinion for academic topics. | If you got 0.274 for Question 1 you appear to be working with the old version of the dataset - a new version was uploaded today and you need to run all the questions again using this dataset and the stepwise should work! |

Table 1 (continued). Coding Schema for BDE datasets

| Category | Definition | Example |
|---|---|---|
| **Software issues (B)** | Concerns about the software (Rapidminer), operating system, Java version, computer language or technical issues that occurred on the course webpage. Technical problems include how to install software or packages and how to use them. | I can't install rapidminer-5.3.013x32- install because "it isn't application Win32" - the message from my Windows |
| **Question about software issues (BQ)** | Questions concerns about the software (Rapidminer), operating system, Java version, computer language or technical issues that occurred on the course webpage. Technical problems include how to install software or packages and how to use them. | I am running OS X 10.7.5. Do I have to upgrade for this? I have Java 7-Update 45, isn't that enough? I have the same problem as Paul. |
| **Announce-ment (A)** | Announcements made by the instructor to post assignments, office hours, due dates, or any official event to tell students information. | Hi everyone. All materials are released Thursdays at noon USA Eastern, so quizzes (assignments really) are also due at that time. (But two weeks later). |
| **Additional materials (M)** | Students provide reading material, papers, or videos, to help their peers understand terms or theories. | Rapid-I have video tutorials on their website http://rapidi.com/content/view/189/212/lang,en/ |
| **Course logistics (C)** | Student answers about the due date of quizzes, discussion about course credit, or some other topic concerned with the course logistic issues. | This issue is still happening, this is quite worrisome...I got a perfect score on the first quiz and don't want to have to redo it in the form of "Assignment 1B" or whatever before the Nov. 7 deadline. I'd prefer not to waste my time. |
| **Question of course logistics (CQ)** | Questions concerned about the due date of quizzes, discussion about course credit, or some other topic concerned with the course logistic issues. | Since no one seems to be jumping in here, I'll start by asking how many weeks of lectures have already been recorded? |
| **Social connection (S)** | Introduction to the class or setting up an online discussion group. | Hello all! I'm Krish, currently doing research in Natural Language Processing at TU Darmstadt, Germany. I am looking forward to learning about EDM and applying it to education in underdeveloped regions. I will also be applying for graduate studies in educational technologies using NLP/Machine Learning. So, hope this course helps! |
| **Politeness (P)** | Show politeness to classmates or instructors, to thank them for their help. | Thank you for adding this! Very helpful! |

Table 1 (continued). Coding Schema for BDE datasets

| Category | Definition | Example |
|---|---|---|
| **Off topic (T)** | Any post that is not relevant to the course, or complaints about the instructor and course. | It helps to download the PDFs and just enjoy his presentation. From personal experience, it's difficult to put personality in videos or audio to begin with, and many people would be complaining if he slowed down and made it more "robotic." However, I do feel your pain. |
| **Other languages (X)** | Posts not in English. | |

Table 2. Distribution of posts types (based on categories described in the coding schema) in BDE datasets

| Sn | Category | BDE2013 | BDE2015 |
|---|---|---|---|
| 1 | On topic statement (O) | 1292 | 228 |
| 2 | Social connection (S) | 335 | 381 |
| 3 | On topic questions (Q) | 666 | 133 |
| 4 | Totally of topic (T) | 494 | 208 |
| 5 | Politeness(P) | 383 | 399 |
| 6 | Software issues (B) | 406 | 362 |
| 7 | Course relevant (C) | 205 | 100 |
| 8 | Question of software issues (BQ) | 201 | 161 |
| 9 | Question of course relevant (CQ) | 105 | 66 |
| 10 | Additional materials (M) | 14 | 10 |
| 11 | Announcement (A) | 9 | 8 |
| 12 | Other languages (X) | 149 | 36 |

## 4.3. Coding schema for JAVA2015 dataset

JAVA2015 dataset was labelled via a survey. The survey consisted of posts from online discussion forum to be rated on a Likert-scale from 1-5 over three parameters namely, urgency, complexity and clarity. Posts for the survey were selected by random sampling from JAVA2015 dataset but keeping out all those posts that didn't require a response. This was essential to make sure that all the posts in the final survey are relevant to the parameters they are rated on. A post that doesn't require a response (e.g. answer posts, announcement posts, etc.) has no relevance for urgency and was hence avoided. The final survey consisted of a total of 150 posts with 146 of them demanding a response (i.e. is an explicit question or expresses confusion that needs to be resolved) while only 4 posts did not necessarily require a response.

Twelve experienced TAs participated in a survey and labeled 50 data instances each along with a justification for their ratings. A total of 150 instances (6% of the dataset) were labelled with each instance labelled by exactly 4 TAs. The final label was determined using the aggregate (mean) of labels from each TA and a cut-off of 3. A label with aggregate rating less than 3 was labelled as non-urgent and greater than or equal to 3 was labelled as urgent in the final dataset that was used in this study. The survey results and findings are discussed in more detail under the following section 4.4 Survey.

## 4.4. Survey

### 4.4.1 Survey Takers:

In order to understand how teaching assistants (or instructors) decide whether a post is urgent or not, I surveyed twelve experienced Teaching Assistants (TAs). Each of these TAs had served as a TA for at least one complete semester before their participation in this study. The lowest experienced TA had an experience of one semester of being a TA, while the most experienced TA had seven semesters of experience and the overall average experience for all TAs surveyed was 2.67 semesters. Five of the surveyed TAs had also served as a TA for at least one semester for the course 'CSC 216 - Programming Concepts - Java' (that I am using the dataset of) for one of its prior iterations.

### 4.4.2. Survey Content:

The survey consisted of posts from the online discussions for the course 'CSC 216 - Programming Concepts - Java' for 2015 offered on campus at NC state University. Each part of the survey constituted of 50 posts to be rated by each TA on a quantitative scale from 1-5 over 3 parameters, namely, urgency, complexity and clarity. I designed 3 such surveys amounting to a total of 150 posts to be rated.

The instructions given to survey taker simply defined the parameters for their understanding, but they were free to rely on their best judgement to rate the posts as they deemed appropriate. Following are the instructions provided to the survey takers defining each parameter that each post had to be rated against:

***Urgency*** - This parameter tries to find - 'How urgent is the post?' or 'How quickly does a TA need to respond to the post?'

***Complexity*** - This parameter tries to find -'How complex is the post to answer?' In other words: can the TA answer the question without referring to any document or resource (Least complex), or do they need to refer to a document or look at a code snippet to answer it (mid-level complex), or do they need to download and execute a code to answer (most complex).

***Clarity*** - This parameter tries to find - 'How clear is the post or how easy is it to understand what is being asked in the post?' In other words, do the TAs need additional information or clarification to understand what is being asked in the post.

Each post had additional metadata information available with it, like time stamp of the post, deadline relating to the post (i.e. if a post is regarding a project/assignment its deadline was also provided) and subject of the post. Also, in cases where the post appeared in the middle of a threaded discussion (i.e. wasn't the first post in the thread) all prior posts in the thread were also included to provide any additional information that might aid the TA in deciding the urgency of the post.

In additional to above 3 parameters, TAs also had to provide a one sentence justification for their chosen urgency rating for each post.

#### 4.4.3. Survey Results:

Each survey taker rated 50 posts on the above described parameters and rating scales, resulted in 150 labelled posts with each post labelled by exactly 4 TAs. The survey resulted in some useful findings and presented me with the opportunity to get into the head of a TA and know how he/she decides the urgency of a post. I used Pearson correlation coefficient to find any existing correlation between mean urgency, complexity and clarity values and the results are tabulated in table 3. Urgency – Complexity and Complexity – Clarity show some correlation while Urgency – Clarity shows little correlation. I believe urgency – complexity correlation can be explained by the fact that most of the urgent posts in this dataset were more detailed with either code snippets or error messages that contributed to high complexity in many cases. Complexity – Clarity correlation is a little more natural as most complex posts tend to be less Clear and hence the relation, also this relation is negative which means clearer a post less complex it is.

Table 3. Pearson Correlation between survey parameters

| Urgency – Complexity | Complexity – Clarity | Urgency – Clarity |
|---|---|---|
| 0.40 | - 0.39 | - 0.162 |

Using the justification provided by TAs for their urgency ratings for each post as a basis, I tried to summarize the reasons for urgency into categories which are as following. (I discarded all those posts that did not provide any clear reason for urgency, and the following distribution is based on 434 posts that were labelled with an unambiguous justification for urgent or non-urgent posts):

Some of the common reasons specified by the TAs for a post to be urgent are:

1. ***Close to deadline***: If a post is regarding an assignment/project that is due soon, TAs view these posts as urgent. Deadline is the most critical factor to decide whether a post is urgent or not. A post is regarded as urgent if it is close to deadline even if it is a silly query because this can directly impact student's grade. 35.78% of the posts tagged as urgent stated *'close to deadline'* as the justification for high urgency.

2. ***Blocking issue***: If a student's doubts are stopping him/her from making significant progress or are blocking him/her in an assignment or project, it is viewed as urgent as the student is stuck and the TA is expected to help. This amounted to 23.28% of all the urgent posts.

3. ***Server issues, or anything that blocks submission***: Any issue with the server or anything that could potentially block assignment/project submissions is viewed as highly urgent. This is specifically urgent for TAs because usually they are the one responsible for maintaining the server and avoiding any other submission issues. This amounted to 7.33% of all the urgent posts.

4. **S*everal students facing same issue***: Queries that are raised by several students is indicative of a potentially widespread confusion and is viewed as urgent. Answering such posts could help other students who may also have the same doubts. This amounted to 6.89% of all the urgent posts.

5. ***Student distressed or frustrated***: Students can often get stuck at something which even after multiple attempts and trying out different approaches is unresolved by them. Student distress is displayed in threads that usually are longer than usual and details multiple failed solutions that they have tried, often these threads have multiple posts by the same student. These posts are viewed as urgent by TAs because the student has tried everything he/she could think of and is still stuck. An urgent assistant and a quick solution may boost up the student's morale and keep him/her going. This amounted to 5.17% of all the urgent posts.

6. ***Ambiguous or incorrect instructions***: Sometimes the instructions for the assignment, exam or project can be ambiguous, inconsistent or incomplete, raising doubts in several student's mind. Any query pointing to a potential issue in the instruction is viewed as urgent as this will impact all student's and needs to be fixed at the earliest. This amounted to 3.88% of all the urgent posts.

7. ***Others***: 17.67% of the posts categorized as urgent stated diverse reasons that doesn't fit into any of the above 6 discussed factors.

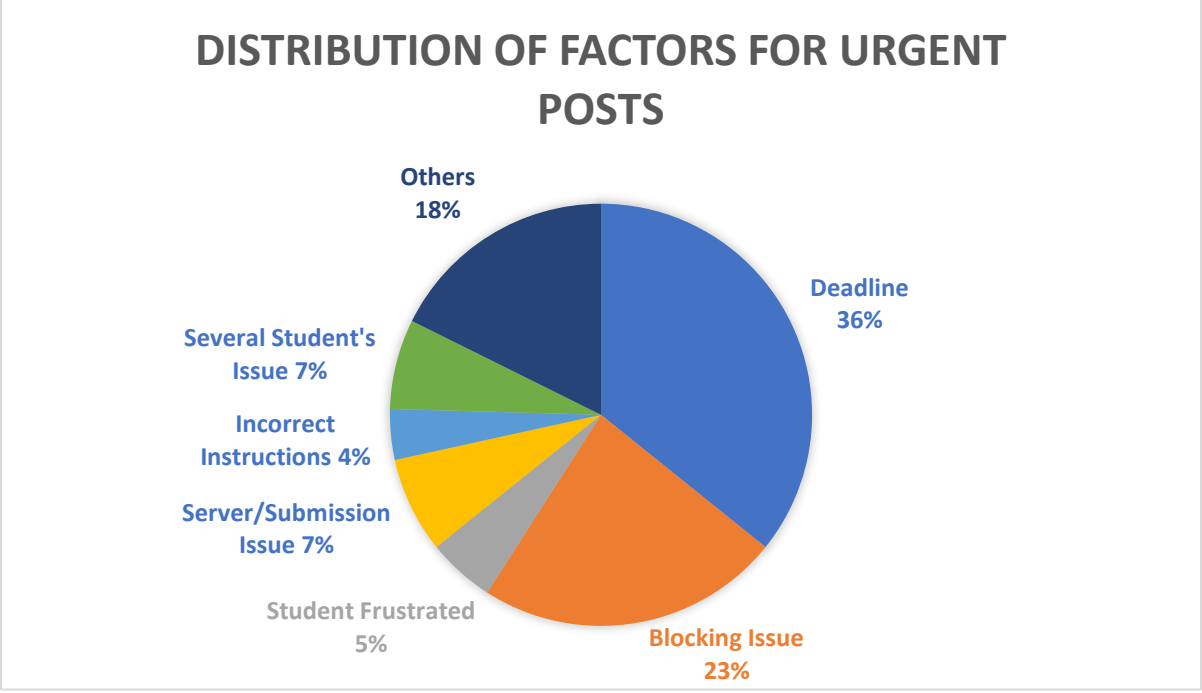


Fig 1. Distribution of Factors for Urgent posts based on the survey results

Some of the common reasons specified by the TAs for a post being non-urgent are:

1. ***Deadline not immediate***: If a post is regarding an assignment/project which isn't due anytime soon, the TAs generally view it as relatively non-urgent. However, this alone doesn't guarantee that the post is not urgent and there are secondary filtering criterions (discussed below) that the TAs use to decide the urgency. In this survey roughly 22.93% of the posts categorized as non-urgent stated solely *'deadline not close'* as a justification.

2. ***Students can figure out themselves***: For all posts that are not concerned with immediate deadlines and can be solved with the information that is already provided to students, the TAs usually take some time to respond. This is just to ensure that the students get enough time to rethink and probably come up with the solution themselves. This amounted to 21.95% of all the non-urgent posts.

3. ***Unimportant/Unnecessary questions***: This includes *'curiosity questions'*, '*doubts that are easy enough for some other student to help with'* and *'doubts that don't directly impact student's grade'*. This amounted to 21.95% of all the urgent posts.

4. ***Non-Blocking issues***: Post concerning issues that need to be resolved but don't necessarily block the student from making progress or working on other parts of the assignment or project are viewed as non-urgent. This amounted to 12.68% of all the non-urgent posts.

5. ***Logistics related queries***: Questions about when the grades will be posted or about grade distributions and other such things are regarded as fairly non-urgent as they are only for student's information and have no potential to impact student's grade and learning. This amounted to 5.37% of all the non-urgent posts.

6. ***Others***: 15.85% of the posts categorized as non-urgent stated diverse reasons that doesn't fit into any of the above 6 discussed factors.

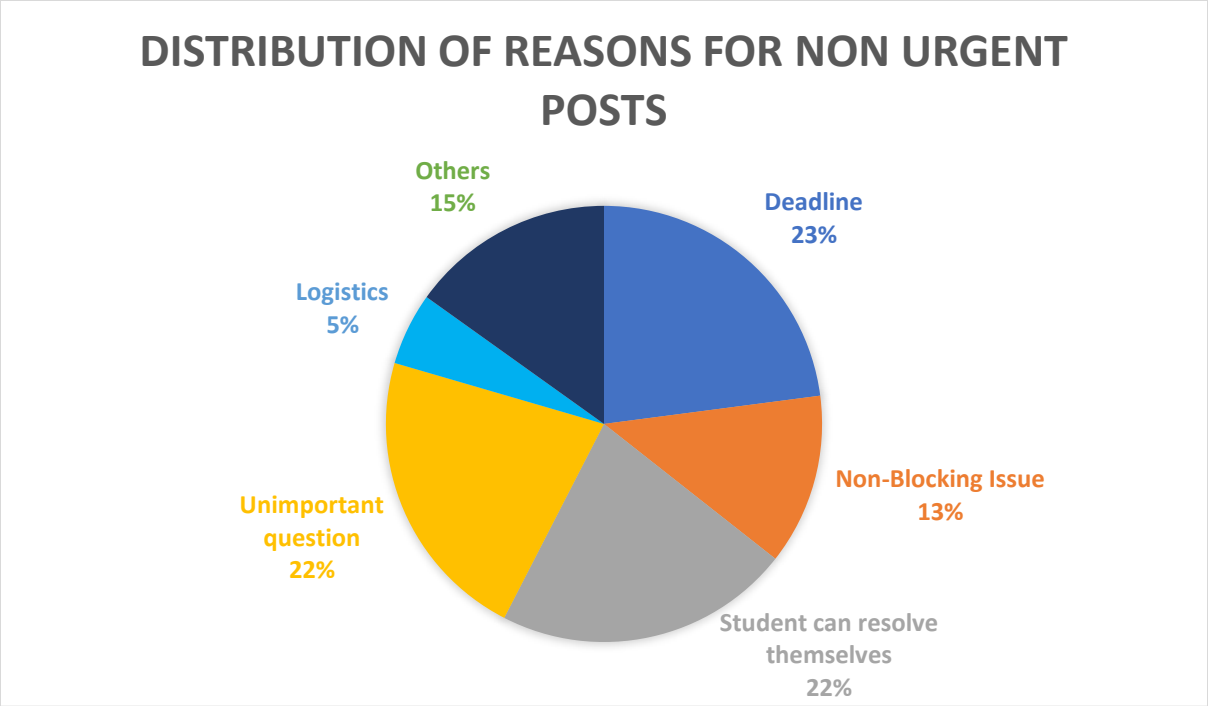


Fig 2. Distribution of Factors for Non-Urgent posts based on the survey results

### 4.5 Interrater/Coding agreement

For BDE datasets, I calculated inter-rater reliability using Cohen's kappa [32] to evaluate the grader and manual reliability. Cohen's kappa is a measurement of agreement between two individuals that also accounts for chance agreement. The kappa value represents the discrepancy between the observed probability of success and the probability of success under the assumption of an extremely bad case. Independence implies that the pair of raters agree about as often as two pairs of people who make random decisions. The maximum value for kappa is 1. Viera et al. provides a set of threshold values for Kappa that are commonly used [32]:

- Poor agreement Less than 0.20
- Fair agreement 0.20 to 0.40
- Moderate agreement 0.40 to 0.60
- Good agreement 0.60 to 0.80
- Very good agreement 0.80 to 1.00

To measure the inter-rater reliability, we randomly selected a 20% data sample from both datasets, and we had three researchers independently label them. We obtained a 0.81 kappa for BDE2013, and 0.71 for BDE2015.

The JAVA2015 was annotated by experienced TAs through the survey. I calculated the interrater agreement for this dataset using Krippendorff's alpha Coefficient [33] as Cohen's Kappa can effectively be used for 2 raters' agreement which was exceeded for this dataset's annotation. Furthermore, Krippendorff's alpha Coefficient is better suited for calculating interrater agreement for annotations based on Likert-scale, as is in this case. The JAVA2015 dataset was annotated on a Likert-scale between 1 and 5, 1 being strongly disagree and 5 being strongly agree. The value for Krippendorff's alpha expresses interrater agreement as following:

$\alpha = 1$ indicates perfect agreement

$\alpha = 0$ indicates perfect disagreement

$\alpha < 0$ when disagreements are systematic and exceed what can be expected by chance.

The Krippendorff's alpha for *urgency*, *complexity* and *clarity* for the entire 150 labelled instances was 0.30, 0.327 and 0.227 respectively. The low agreement between raters for all three parameters demonstrates there is low consensus as to what is an urgent post, or how clear or complex a post is for individual TAs. The Likert-scale for rating these parameters further added to low agreement as different TAs differ in how they perceive the scale and the rating, and little directions were provided regarding the scale to ensure fair results from the survey. This is also demonstrated by the distribution of factors determining urgency in section 4.4.3 which discussed the distribution of factors for deciding whether a post is urgent or not. There are a number of factors to take into account to decide the urgency of a post and each TA weighs these factors differently. Some TAs believe that an early response to student's doubts can boost their morale and help them make progress while some believe giving them more time to think on issues that can be resolved easily could contribute to better understanding of the course content and their individual growth in the course.

In spite of the low agreement as each instance has been labelled by 4 TAs, averaging the individual ratings could yield a much better result that is close to the actual label. Assuming this to be true, I averaged ratings for each label and categorized any post with mean rating lower than 3 as non-urgent and greater than or equal to 3 as urgent posts. This resulted in a distribution of {Non-urgent: 57, Urgent: 93} of the 150 labelled instances. These labels were used to train the semi-supervised model along with the other 2509 unlabeled instances from the same dataset.

## 4.6 Data preprocessing

Data preprocessing differed slightly for the BDE2015 and BDE2013 as compared to JAVA2015, primarily because of the inherent difference in the characteristics of each dataset. In addition to standard data preprocessing for NLP, JAVA2015 required additional custom preprocessing. Following sections describe the data-preprocessing in more detail.

### 4.6.1. Data preprocessing (phase 1)

This data preprocessing was applied to all 3 datasets alike. This includes some standard data preprocessing to make the data better suited for classification.

***Stop-word removal:***

Many words in a sentence are very frequent/redundant but convey little information. These are called stop words. These are the words needed to support the grammatical structure of a language but beyond that have no more use. Such words are noise for classification process and are better off removed before classification. Some of the English language stop-words are – the, a, but, in, or, so, etc. These words can be removed from the text without losing much of the meaning of the actual text.

***Remove punctuation:***

Punctuations add little to no meaning for the purpose of text classification and are hence removed from all datasets, except for question marks. For this study of finding urgency in posts (which is closely related to explicit questions in posts) question marks are particularly important and all questions marks were replaced with a custom tag <Q_MARK> in all 3 datasets.

***Remove digits and numbers (Numerals):***

Numbers add little value for the classification task at hand and hence, all digits and numbers were replaced with a tag <DIGIT> in the datasets to have more uniformity across all three datasets.

***Lemmatization:***

Lemmatization is a morphological analysis of the words that captures canonical forms based on a word's lemma. For example, the two words 'better' and 'good' are very similar in meaning and if treated alike could enhance the classification performance that takes vectors based on word counts and distributions. Lemmatization of word better would result in the following:

better → good

***Stemming:***

Stemming is the process of eliminating affixes (suffixes, prefixes, infixes, circumfixes) from a word in order to obtain a word stem. Often different variations of the same word like, go-going, run-running etc., exists in text. These subtle differences could be useful in some NLP tasks, however for our purpose of classification, collapsing these similar words into one could enhance the performance.

running → run

### 4.6.2. Data preprocessing (phase 2)

The JAVA2015 dataset being a programming course had lots of programming language specific information in the posts. As the course is a programming/coding course most of the assignments/projects were coding related and students were required to submit their working code for most assignments. Students would often post their doubts with URL links to GitHub repository or just paste their code or error message in the discussion posts. This information though extremely useful for the discussion possesses little value for our task at hand. Hence, I decided to collapse all this information in the post and replace it with custom tags. This custom data preprocessing isn't required for BDE2013 and BDE2015 as these courses do not have any idiosyncrasy like JAVA2015 dataset.

***i. Code blocks:***

A specific challenge in identifying a code block in online posts is to identify the beginning and the end of the code as often these code snippets are preceded and succeeded by text in a single post explaining the student's doubts. Students don't necessarily follow the rules of the programming language (like appropriate opening and closing brackets) while posting their code in discussion forums. The code snippet could be a method block, a class block or just a few lines from the middle of some code. All this added to the problem and using a simple regex alone wasn't enough. I designed a simple yet effective string parser along with regex to identify code snippets in the posts. This was specifically designed for Java code as the all code snippets in JAVA2015 dataset are of Java and was effective enough to identify most of the code snippets in the given dataset. A possible and more effective alternative would be to use machine learning to identify code snippets in the text, but I couldn't explore that in this work.

Code snippets were present in myriads of format and were replaced with following custom tags:

i. <CLASS_BLOCK>: This included code that was inside a class and may include more than one method within it.
ii. <METHOD_BLOCK>: This included code that was part of a method/subroutine. Method blocks in this dataset could appear without the class information, however if a method is present within a class, the <CLASS_BLOCK> will take precedence.
iii. <METHOD_DECLERATION>: This included all those codes that simply referred to some method or that declared method names but not the entire method block in the discussion post.

***ii. File paths and URL links:***

File paths and URL links were identified using regex and were replaced with custom tags <FILE_PATH> and <WEBSITE_ADDRESS> respectively.

***iii. Error messages:***

Error messages are a common occurrence in JAVA2015 dataset as most assignments and projects were coding based. Error messages were identified using regex and were replaced with the tag <ERROR_MESSAGE>.

***iv. Comment lines:***

Comment lines in the dataset were either single line or multiple lines and were identified using regex. I used custom tag <COMMENT_LINE> to replace all comments from the dataset, multi-line comments were collapsed and replaced with a single tag.

The phase 2 of data pre-processing made the JAVA2015 dataset more like the BDE2013 and BDE2015 datasets to facilitate effective use of models trained on those datasets. Further, code blocks, error messages, comments, and URLS formed a huge chuck of text in the dataset but did not add any useful information for our purpose. Replacing these texts with custom tags makes the dataset more uniform such that all code snippets are replaced with a specific tag across the dataset, and similarly for other above-mentioned entities that were replaced with custom tags. Though these simple techniques were not robust enough to identify all occurrences of these entities but was good enough to identify most of it.

### 4.7 Lexical features and metadata

As an input to my models I used Tf-idf vectorization for text, metadata information like deadlines, and some custom features.

***Tf-idf***: Term frequency-inverse document frequency (Tf-idf) [34] is a weighting method that is used to estimate the relative importance of a term (word/words) in a given document by dividing its relative frequency within the post by its relative frequency within the entire corpus. Tf-idf is a ranking metric that was developed for information retrieval applications and is frequently used in text summarization and classification.

***Deadlines***: JAVA2015 dataset has timestamps for each post and in addition also has deadline information available for posts that concern a deadline. The difference between timestamp and deadline is the time remaining before the deadline of the assignment or project that the post is regarding. I normalized the value of this difference using the concave down curve (fig. 4.6.1) and used this information as a feature to my model. Any posts with more than 7 days remaining for the deadline gets a deadline weight as 0, and for all other differences within 7 days the weight is

calculated using the curve (fig. 4.6.1). I used this curve (fig 4.6.1) and not a linear function to calculate deadline weight because the survey results indicated deadlines as being a depreciating factor (not linear but concave down) for urgency with increased time. The cut-off value of 7 days was decided based on the fact that posts with more time than 7 days before deadline were not reported urgent based on deadline and hence deadline doesn't play a role in those posts, so a weight 0.

***Question marks***: Previous studies have found that in many cases, question marks are a very important punctuation feature when identifying question sentences [35] [36] [37]. So, I included the question mark as a separate feature for the classification model.

***Custom word features***: Custom word features such as 'what', 'where', 'how', 'why', 'which' and 'who' can be crucial in identifying questions in text. So, I included these custom word features as a separate feature for the classification model.

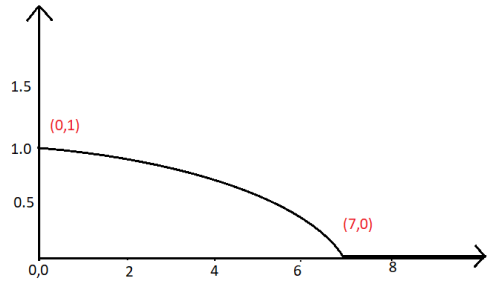


Fig. 3. Concave down curve to calculate the deadline weights. The curve follows the equation $y = 1 - 0.02 * x^2$

CHAPTER

# 5

# IMPLEMENTATION

## 5.1 Supervised Model

In this study, I evaluated five different classification algorithms, namely, Support Vector Machines, Random Forest, Decision Trees, Logistic Regression and K-Nearest Neighbors. I first trained the classifiers on BDE2013 for binary classification to categories posts into ‘questions’ and ‘non-questions’ and tested the performance on BDE2015 dataset to test cross-domain classification for this model. Further I trained these classifiers on combined BDE2015 and BDE2013 datasets and labelled samples from JAVA2015 dataset to classify unlabeled posts in JAVA2015 into ‘questions’ and ‘non-questions’.

## 5.2. Partially Supervised Model

Also known as one-class classification or unary classification, this is a class of machine learning algorithms that tries to identify objects of a specific class based on training from only labelled instances of that class. It uses only sample points from the assigned class, so that a representative sampling is not strictly required for non-target classes. One-class SVM [38] is based on the idea of unsupervised outlier detection that identifies outlier data as representative of the second-class.

The model is trained using only positive and unlabeled instances and can be used to identify positive instances in test data.

In this study I used one-class SVM to identify questions in unlabeled JAVA2015 data. Identifying questions is critical for this study as once a post is identified as being a non-question it can either be given a lower weight or discarded before trying to find its urgency. Non-questions are posts that do not necessarily demand a response and hence can be treated as non-urgent.

### 5.3 Semi Supervised Model

For semi-supervised models I used Expectation Maximization and Label-spreading to train on both labelled and unlabeled data from JAVA2015 dataset.

*Expectation Maximization* [39] is a statistical iterative method that calculates maximum likelihood or maximum a posteriori (MAP) estimates using latent variables in labelled and unlabeled instances. The iterations constitute of 2 steps, namely, Expectation step (E) and a Maximization step (M). Expectation step creates a function for the expectation of the log-likelihood that is evaluated using the current estimate for the parameters. Maximization (M) step computes parameters maximizing the expected log-likelihood found on the step E and uses these estimates to determine the distribution of the latent variables in the next step E. For this study I used Naïve Bayes Classifier with multinomial distribution and expectation-maximization algorithm to optimize the results.

*Label Spreading* [40] is an unregularized graph based semi-supervised learning algorithm that uses community structure emergence in labelled and unlabeled instances to determine class membership. Starting with a random distribution the algorithm tries to assign labels to each instance based on other labeled instances close to it in the graphical network in a manner analogous to K-Nearest-Neighbor (KNN) in traditional supervised learning. It uses affinity matrix based on the normalized graph Laplacian and soft clamping across the labels. Label spreading is based on spreading activation networks in which every point iteratively spreads its label information to its neighbors until a global stable state is achieved.

## 5.4 Final Combined Model

I designed two slightly different models by combining the supervised, partially-supervised and semi-supervised learning models to predict and classify posts as either urgent or non-urgent. The primary structure and logic of each of these models is the same but they differ in the way they handle the predicted labels from the supervised model and partially-supervised models before utilizing the semi-supervised models for final prediction.

### 5.4.1 Model 1 (Filter out non-questions)

Fig. 4 shows model 1a and explains the data-flow and working of it in some detail. Classifier_1 is a supervised model trained on combined BDE2013, BDE2015 and Java2015 (labelled instances). It is a binary classifier that classifies posts as either posts needing response (i.e. questions) and posts not needing response (i.e. non-questions). This trained model is then used on the unlabeled instances in Java2015 dataset to classify as posts needing response and posts not needing response. All posts that are classified as not needing response are then discarded and only the posts needing response are kept for further training. An urgent post is one that demands a response as quick as possible, a post that doesn't need a response can thus not be urgent and no useful information is lost on discarding such posts. The filtered posts (still unlabeled for urgency) are then combined with labelled instances (150 instances labelled for urgency) to train a second learning model which is a semi-supervised classification model. This trained classifier, classifier_2 can be used to classify posts into urgent and non-urgent.

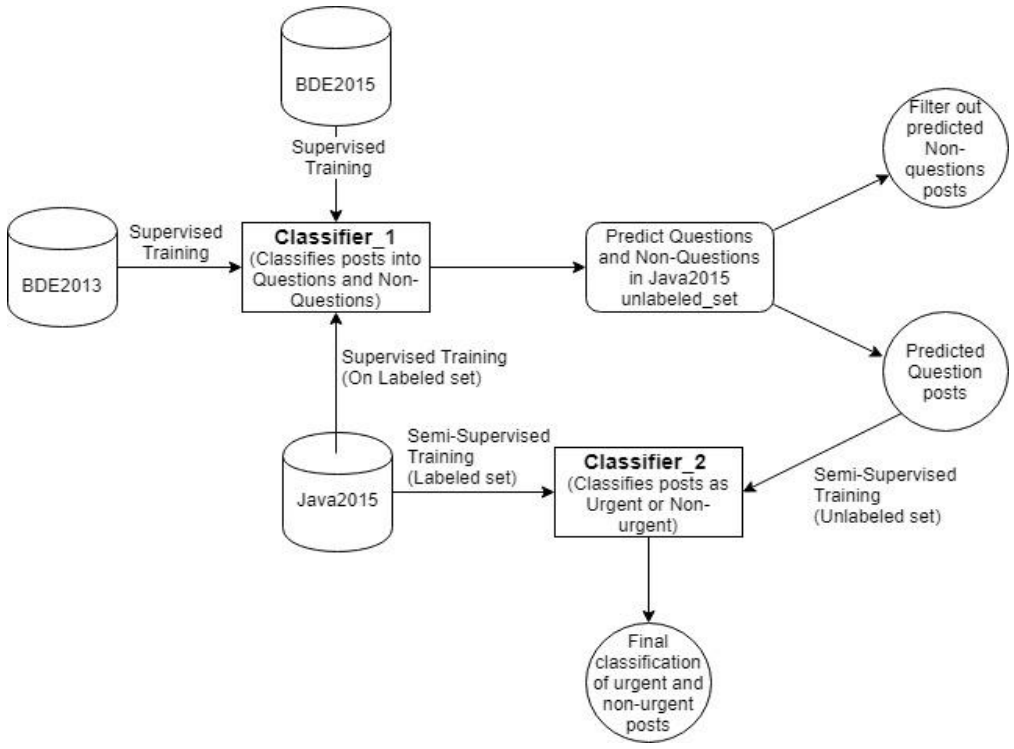


Fig 4. Model 1a. Classifier 1 is a supervised model trained on combined BDE datasets and labeled instances in JAVA2015 dataset. Classifier 2 is a semi-supervised model trained on filtered unlabeled instances of JAVA2015 from classifier 1 and labelled instances from JAVA2015 dataset. Final Prediction for urgency of the post is predicted by classifier 2.

A slightly modified version of this model shown in Fig. 5 is one in which I replaced supervised learning classifier_1 with partially supervised learning classifier trained only on JAVA2015 labelled instances, fig. 5 shows this model 1b. One major difference that this slight variation introduces is that both partially supervised and semi-supervised models are trained on the same dataset, hence, eliminating the need of a tagged dataset for training a supervised model as in model 1a.

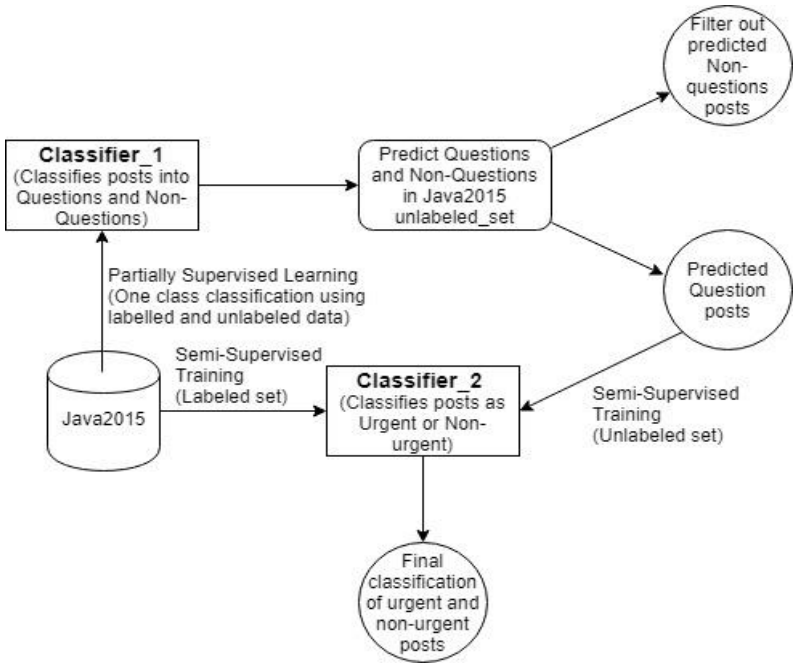


Fig. 5 Model 1b. Classifier 1 is a partially supervised (one-class SVM) model trained on labeled instances in JAVA2015 dataset. Classifier 2 is a semi-supervised model trained on filtered unlabeled instances of JAVA2015 from classifier 1 and labelled instances from JAVA2015 dataset. Final Prediction for urgency of the post is predicted by classifier 2.

### 5.4.2. Model 2 (Use question/non-question prediction as an input feature)

Fig 6 shows model 2a and explains the dataflow and working of it in some detail. Classifier_1 is trained exactly as explained for model 1a with no change. The difference in this model is how the predicted labels from classifier_1 are handled. Rather than discarding any data instances based on classifier_1 predictions, these predictions are passed on as input features for classifier_2. Classifier_2 is trained on combined labelled and unlabeled data instances and uses predictions from Classifier_1 as additional input feature for training. Trained Classifier_2 can then be used to predict urgent and non-urgent posts.

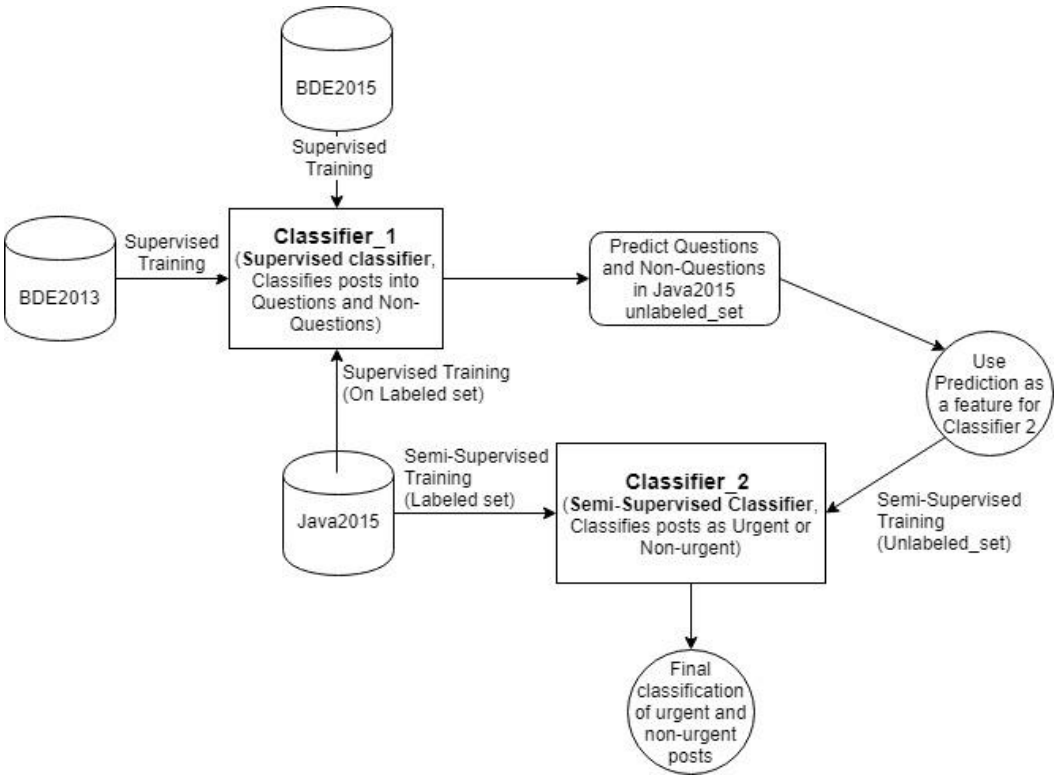


Fig 6. Model 2a. Classifier 1 is a supervised model trained on combined BDE datasets and labeled instances in JAVA2015 dataset. Classifier 2 is a semi-supervised model trained on unlabeled instances of JAVA2015 combined with prediction from classifier 1 as an added feature for each unlabeled instance. Final Prediction for urgency of the post is predicted by classifier 2.

Just like Model 1b, model 2b (fig. 7) is a slight variation of the Model 2a that needs only one dataset that can be used for training both classifiers (a one-class SVM), and a semi-supervised classifier for final prediction.

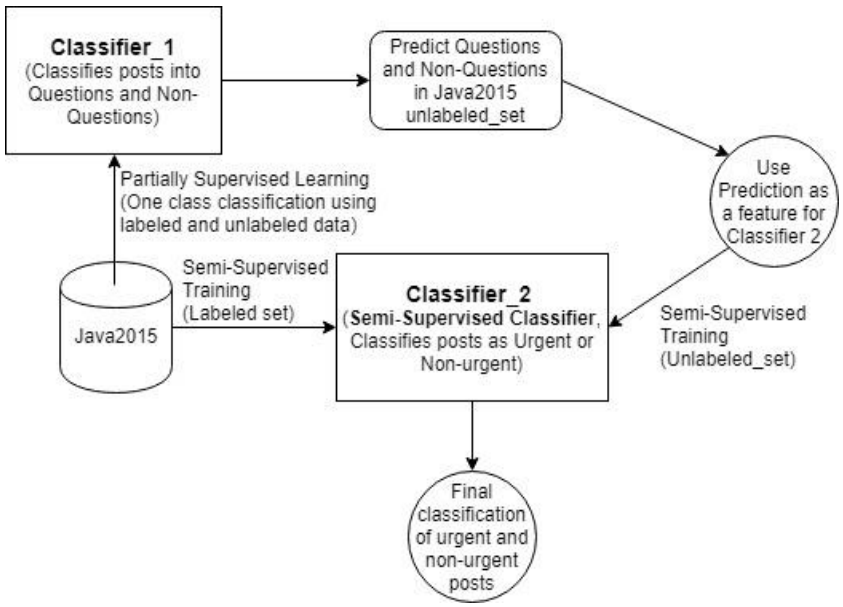


Fig 7. Model 2b. Classifier 1 is a partially supervised (one-class SVM) model trained on labeled instances in JAVA2015 dataset. Classifier 2 is a semi-supervised model trained on unlabeled instances of JAVA2015 combined with prediction from classifier 1 as an added feature for each unlabeled instance. Final Prediction for urgency of the post is predicted by classifier 2.

CHAPTER

# 6

# RESULTS

I first experimented with semi-supervised classification models alone. Trained on 150 labelled instances and 2509 unlabeled instances from the JAVA2015 dataset and validated using 10-fold stratified cross-validation on the labelled instances. Table 4 shows results for the vanilla semi-supervised models.

Table 4. Results for semi-supervised models trained on labelled and unlabeled instances in JAVA2015 dataset. Validation was performed using the 150 labelled instances with 10-fold stratified cross-validation. 'P' is Precision, 'R' is Recall and 'F1' is F1 score

| **n-grams** | **Semi-supervised Classifier** | **Urgent** | | | **Non-Urgent** | | | **Overall Accuracy** |
|---|---|---|---|---|---|---|---|---|
| | | **P** | **R** | **F1** | **P** | **R** | **F1** | |
| Unigram | Expectation Maximization | 0.57 | 0.80 | 0.66 | 0.29 | 0.12 | 0.17 | 0.52 |
| Bigram | Expectation Maximization | 0.61 | 0.79 | 0.69 | 0.49 | 0.28 | 0.36 | 0.58 |
| Unigram | Label Spreading | 0.59 | 0.79 | 0.67 | 0.40 | 0.20 | 0.27 | 0.55 |
| Bigram | Label Spreading | 0.59 | 0.98 | 0.74 | 0.50 | 0.03 | 0.06 | 0.59 |

Both semi-supervised classifiers performed averagely with bigram models for both models performing slightly better than the unigram models. Performance for both classifiers are comparable, with Label spreading marginal better in terms of accuracy, however, the better performing label spreading classifier is heavily biased towards one-class (urgent in this case). Same is true for all other results in semi-supervised models with varying degree. Due to this bias,

the semi-supervised classifiers alone are not sufficient to reliably classify the posts into urgent and non-urgent.

I then experimented with one-class SVM classifier and tested its performance on the 150 labelled instances. I trained the one-class SVM to identify question posts in the unlabeled instances. Table 5 shows results for various n-gram one-class svm models on JAVA2015 dataset, the model is trained on 146 positively labelled posts (i.e. questions) and the results are from 10-fold stratified cross-validation over 146 labelled instances. The classifier in general performed poorly which I think is because of the low number of training instances (146 instances).

Table 5. Results for one-class SVM trained on labelled instances in JAVA2015 dataset. Validation was performed using the 150 labelled instances with 10-fold stratified cross-validation. The results only have accuracies because it is a one-class prediction.

| **Model** | **1-gram** | **2-gram** | **3-gram** | **4-gram** |
|---|---|---|---|---|
| Accuracy | 0.47 | 0.53 | 0.56 | 0.58 |

Combining this one-class SVM classifier with semi-supervised model as described in Fig. 5 and 6 (models 1b and 2b), did improve the overall performance noticeably. Table 6 shows the results for the model 1a and 2a that uses one-class SVM as classifier 1 and expectation maximization and Label Spreading (separately) for classifier 2. The results show the model 1b enhanced the performance 2% for expectation maximization and by 9% for label spreading compared to the best performing vanilla semi-supervised models alone (table 4), and the model 2b showed 5% improvement for both EM and LS in terms of accuracy. The major difference in these models compared to simple semi-supervised models is that these models are much less biased towards one-class and the results are above 50% for all the metrics, namely, precision, recall, F1 score and accuracy (except for recall and F1 score for non-urgent class for model 2b with EM).

Table 6. Results for models 1b and 2b for unigram models. One-class classifier is trained on labelled instances in JAVA2015 while Classifier 2 is trained on combined labelled and unlabeled instances from JAVA2015. 'P' is Precision, 'R' is Recall and 'F1' is F1 score.

| **Combined Model** | **Classifier 1** | **Classifier 2** | **Urgent** | | | **Non-Urgent** | | | **Overall Accuracy** |
|---|---|---|---|---|---|---|---|---|---|
| | | | **P** | **R** | **F1** | **P** | **R** | **F1** | |
| Model 1b | One SVM | EM | 0.67 | 0.66 | 0.67 | 0.52 | 0.53 | 0.53 | 0.61 |
| Model 1b | One SVM | LS | 0.71 | 0.77 | 0.74 | 0.62 | 0.55 | 0.58 | 0.68 |
| Model 2b | One SVM | EM | 0.63 | 0.92 | 0.75 | 0.67 | 0.23 | 0.35 | 0.64 |
| Model 2b | One SVM | LS | 0.69 | 0.72 | 0.70 | 0.57 | 0.53 | 0.55 | 0.64 |

Adding a classifier to identify questions before trying to determine the urgency of the posts evidently boosts the overall performance, as shown by the results in table 6. To experiment further and improve the performance I trained a supervised classifier on the BDE2015 and BDE2013 datasets to categorize posts into questions and non-questions. I experimented with five popularly used supervised classifiers, namely, Support vector machines, Logistic Regression, Random Forest, Decision Trees and K-Nearest Neighbors.

I first trained these classifiers on BDE2013 and tested on BDE2015 to determine the cross-domain classification performance across these 2 MOOC datasets. The model in general performed decently and the tri-gram models performed better in general for all approaches compared to unigram and bigram models, table 7 show results for the cross-domain classification task.

Table 7: Results are for trigram supervised classification models trained on BDE2013 and validated on BDE2015. 'P' is Precision, 'R' is Recall and 'F1' is F1 score.

| **Classifier** | **Questions** | | | **Non-Questions** | | | **Overall Accuracy** |
|---|---|---|---|---|---|---|---|
| | **P** | **R** | **F1** | **P** | **R** | **F1** | |
| logistic regression | 0.73 | 0.66 | 0.69 | 0.92 | 0.94 | 0.93 | 0.89 |
| Decision Tree | 0.56 | 0.61 | 0.58 | 0.90 | 0.88 | 0.89 | 0.83 |
| Random Forest | 0.67 | 0.48 | 0.56 | 0.88 | 0.94 | 0.91 | 0.85 |
| Linear SVM | 0.71 | 0.72 | 0.72 | 0.93 | 0.93 | 0.93 | 0.89 |
| K-Nearest Neighbors | 0.73 | 0.46 | 0.57 | 0.88 | 0.96 | 0.92 | 0.86 |

Then I trained these classification models on combined dataset BDE2013 and BDE2015 to test on the 150 labeled instances from Java2015 dataset. The 150 labeled instances in Java2015 are split as {Questions:146, Non-Questions:4}, so the metric for non-question fluctuated heavily (given there were only 4 labels to be classified and even 1 misclassification would greatly impact the metrics for this class). But in general, the classifiers performed decently. Table 8 shows the results for this cross-domain classification. With the exception of 'Random Forest' all other classifiers performed comparable to cross-domain classification performance between the 2 BDE datasets (table 7).

Table 8. Results are for trigram supervised classification models trained on combined BDE2013 and BDE2015 datasets and validated on JAVA2015 labelled instances using 10-fold stratified cross-validation. 'P' is Precision, 'R' is Recall and 'F1' is F1 score.

| **Classifier** | **Questions** | | | **Non-Questions** | | | **Overall Accuracy** |
|---|---|---|---|---|---|---|---|
| | **P** | **R** | **F1** | **P** | **R** | **F1** | |
| logistic regression | 0.99 | 0.95 | 0.97 | 0.30 | 0.75 | 0.43 | 0.95 |
| Decision Tree | 1.00 | 0.84 | 0.91 | 0.15 | 1.00 | 0.26 | 0.85 |
| Random Forest | 1.00 | 0.38 | 0.55 | 0.04 | 1.00 | 0.08 | 0.39 |
| Linear SVM | 0.99 | 0.88 | 0.93 | 0.15 | 0.75 | 0.25 | 0.88 |
| K-Nearest Neighbors | 0.99 | 0.79 | 0.88 | 0.09 | 0.75 | 0.16 | 0.79 |

I combined the supervised classifier with the semi-supervised classifiers using 2 different approaches explained in fig. 4 for model 1a and fig. 6 for model 2a under the section 5.4.1 and 5.4.2. The results for model 1a are tabulated in table 9 and 10 and results for model 2a are in table 11 and 12.

Table 9. Results are for model 1a using trigram models. Various supervised models (classifier 1) trained on BDE datasets and Expectation maximization (EM) (classifier 2) is a semi-supervised classifier trained on labelled and unlabeled data from JAVA2015 dataset.

| **Classifier 1** | **Classifier 2** | **Urgent** | | | **Non-Urgent** | | | **Overall Accuracy** |
|---|---|---|---|---|---|---|---|---|
| | | **P** | **R** | **F1** | **P** | **R** | **F1** | |
| logistic regression | EM | 0.71 | 0.66 | 0.68 | 0.50 | 0.56 | 0.53 | 0.62 |
| Decision Tree | EM | 0.74 | 0.66 | 0.70 | 0.53 | 0.63 | 0.58 | 0.65 |
| Random Forest | EM | 0.72 | 0.74 | 0.73 | 0.56 | 0.53 | 0.54 | 0.66 |
| Linear SVM | EM | 0.71 | 0.60 | 0.65 | 0.48 | 0.60 | 0.53 | 0.60 |
| K-Nearest Neighbors | EM | 0.69 | 0.62 | 0.66 | 0.47 | 0.54 | 0.50 | 0.59 |

Table 10. Results are for model 1a using unigram models. Various supervised models (classifier 1) trained on BDE datasets and Label Spreading (LS) (classifier 2) is a semi-supervised classifier trained on labelled and unlabeled data from JAVA2015 dataset.

| **Classifier 1** | **Classifier 2** | **Urgent** | | | **Non-Urgent** | | | **Overall Accuracy** |
|---|---|---|---|---|---|---|---|---|
| | | **P** | **R** | **F1** | **P** | **R** | **F1** | |
| logistic regression | LS | 0.65 | 0.71 | 0.68 | 0.45 | 0.39 | 0.42 | 0.59 |
| Decision Tree | LS | 0.64 | 0.70 | 0.67 | 0.43 | 0.37 | 0.40 | 0.57 |
| Random Forest | LS | 0.65 | 0.69 | 0.67 | 0.44 | 0.40 | 0.42 | 0.58 |
| Linear SVM | LS | 0.64 | 0.68 | 0.66 | 0.41 | 0.37 | 0.39 | 0.56 |
| K-Nearest Neighbors | LS | 0.65 | 0.72 | 0.68 | 0.45 | 0.37 | 0.40 | 0.59 |

Table 11. Results are for model 2a using bigram models. Various supervised models (classifier 1) trained on BDE datasets and Expectation Maximization (EM) (classifier 2) is a semi-supervised classifier trained on labelled and unlabeled data from JAVA2015 dataset.

| **Classifier 1** | **Classifier 2** | **Urgent** | | | **Non-Urgent** | | | **Overall Accuracy** |
|---|---|---|---|---|---|---|---|---|
| | | **P** | **R** | **F1** | **P** | **R** | **F1** | |
| logistic regression | EM | 0.66 | 0.63 | 0.65 | 0.44 | 0.47 | 0.46 | 0.57 |
| Decision Tree | EM | 0.65 | 0.59 | 0.62 | 0.42 | 0.49 | 0.46 | 0.55 |
| Random Forest | EM | 0.69 | 0.63 | 0.66 | 0.47 | 0.53 | 0.50 | 0.59 |
| Linear SVM | EM | 0.66 | 0.63 | 0.65 | 0.44 | 0.47 | 0.46 | 0.57 |
| K-Nearest Neighbors | EM | 0.67 | 0.63 | 0.65 | 0.45 | 0.49 | 0.47 | 0.58 |

Table 12. Results are for model 2a using tri-gram models. Various supervised models (classifier 1) trained on BDE datasets and Label Spreading (LS) (classifier 2) is a semi-supervised classifier trained on labelled and unlabeled data from JAVA2015 dataset.

| **Classifier 1** | **Classifier 2** | **Urgent** | | | **Non-Urgent** | | | **Overall Accuracy** |
|---|---|---|---|---|---|---|---|---|
| | | **P** | **R** | **F1** | **P** | **R** | **F1** | |
| logistic regression | LS | 0.65 | 0.92 | 0.76 | 0.59 | 0.18 | 0.27 | 0.64 |
| Decision Tree | LS | 0.64 | 0.91 | 0.75 | 0.53 | 0.16 | 0.24 | 0.63 |
| Random Forest | LS | 0.64 | 0.90 | 0.75 | 0.53 | 0.18 | 0.26 | 0.63 |
| Linear SVM | LS | 0.64 | 0.91 | 0.76 | 0.56 | 0.18 | 0.27 | 0.63 |
| K-Nearest Neighbors | LS | 0.65 | 0.92 | 0.76 | 0.59 | 0.18 | 0.27 | 0.64 |

As the results show among models 1a and 2a, the model 1a using Expectation Maximization was the best performing model overall, with least bias towards either class and a relatively good and consistent accuracies for all supervised classifiers. Model 2a with Label Spreading achieved better accuracy measures for most supervised classifiers but the results are heavily biased towards one class (Urgent class) and the all metrics for non-urgent class are low. Model 2a with expectation Maximization also gave decent results (table 11).

CHAPTER

7

# DISCUSSION

This research investigated the application of semi-supervised learning and other machine learning models combined to prioritize online student discussion posts. The goal of this study was to examine the possibility of effectively training a model with very limited labeled instances using linguistic features and some metadata information. The urgency of a post is viewed from an instructor's point of view as the labels were collected from experienced teaching assistants.

The semi-supervised model alone trained on 150 labeled and 2509 unlabeled instances performed poorly in general mostly because of being biased towards one class. The classifier predicts most instances as majority class which boosts the metrics for one class at the cost of the other. Combining one-class SVM with the semi-supervised classifiers improved the results noticeably and also balanced the predictions for both classes. Label Spreading performed marginally better than Expectation Maximization with one-class SVM for both models 1b and 2b. Combining supervised-models trained on MOOC datasets also improved the performance over semi-supervised models when used alone. However, for model 1a and 2a, Label Spreading which achieved good accuracy measures showed some bias towards the majority class in most cases while Expectation Maximization performed decently in terms of metrics for both classes.

Model 1b with Label spreading was the best performing model overall with Model 1a with Expectation Maximization trailing slightly behind. Semi-supervised classifiers when used alone showed worst performance. Hence, it can be concluded that combining partially supervised models (model 1b and 2b) and supervised models (models 1a and 2a) with semi-supervised model did contribute to improving the overall model's performance.

The research questions established in section 3.1 can now be answered based on these results.

*RQ1. What is urgency with regards to student discussion posts? How do teaching assistants and instructors view posts and what factors do they take into account in deciding whether a post is urgent or non-urgent?*

As the survey results discussed in section 4.4 show, there is little consensus amongst the TAs as to what determines the urgency of a post. Fig. 1 and 2 shows the distribution of various factors that are taken into account by TAs in deciding whether a post is urgent or not, however, each TA weighs these factors differently. As a result, there is little agreement amongst TAs as to what an urgent post is, this is reflected in low interrater agreement in the surveyed samples. A further study is needed in a more controlled research setup to answer this question. Some prior training for the survey takers to establish a standard common scale for ratings can yield results that can be better generalized and has better interrater agreement.

*RQ2. Can textual features in the post and some metadata reliably identify urgent posts?*

The results for different approaches show an overall average performance in predicting urgency of a post, however, I believe there is scope for improvement. First, due to the limited labelled instances the dataset wasn't very well separable, and the low interrater agreement further deteriorated the performance. I would like to investigate more approaches to address these issues in my future work. But there are a number of factors that are considerably difficult to capture using linguistic features and metadata that the human TAs use in determining urgency of a post. For instance, finding whether a post is 'Blocking or non-blocking' or identifying 'unimportant questions' which were stated as major deciding factors in determining the urgency of a post by TAs, are considerably difficult to capture using linguistic features and metadata alone. Some

insights and knowledge of the course and assignments would be necessary to determine these factors. In the light of all the results I conclude linguistic features and basic metadata information about the posts alone are not sufficient to determine the urgency of a post reliably.

*RQ3. Can we train a robust model on limited labelled instances and achieve good performance? (150 labelled instances as used in this study)*

As my research demonstrates very limited labels is too less to effectively train a robust model for a classification. This I think is primarily because limited labels do not comprehensively capture enough vocabulary for effective classification over all other posts. Further, this research focused on finding urgency among all posts that requires a response, which is challenging if our primary source of feature is text. Two posts that have very similar vocabulary and asking similar questions can have very different urgency based on other factors such as, close deadlines, blocking issue for the student, or its impact on several students etc., factors that the TA ranked high in determining the urgency but are very difficult to capture from the text alone.

*RQ4. Can classifiers trained on one MOOC course be effectively used on other MOOCs and blended courses?*

Results in the table 7 and 8 demonstrate a decent performance by most classifiers on cross-domain classification in classifying posts into questions and non-questions. The performance is better when the 2 courses are related or similar, as in table 7 which represents results for model trained on BDE2013 and tested on BDE2015, while we notice some decline in performance when trained on BDE datasets and tested on JAVA2015. JAVA2015 dataset is a blended course dataset while BDE is a MOOC dataset, hence they differ in the types of post they have. For instance, there are far more socializing and politeness posts in a MOOC dataset compared to blended course dataset. This is because students in blended course have on-campus classes and can meet the other students and instructor (or TAs) in class, while the same is not true for MOOC students who often find discussion forum as the only source of communication among students and between instructor and students. But, in spite of these differences the classifier performed reasonably well for cross-domain classification to identify questions across MOOC and blended course dataset which is an interesting finding of this research study.

CHAPTER

8

# LIMITATIONS AND FUTURE WORK

## 8.1 Limitations

There are three major limitations of this research study:

The survey was designed using posts from one dataset and consisted of randomly sampled 150 posts which by no measure can be considered well representative of the entire dataset. Hence, the generalization of survey results like: distribution of factors influencing urgent and non-urgent posts discussed under section 4.4.3 can't be considered universal. At best we can expect similar results to hold for other blended course datasets but the distribution for each factor might change depending on the sample used for the survey.

Another limitation related to the performance of the models proposed are that they are trained on labels attained by averaging ratings from the survey. But the interrater agreement among survey takers is pretty low (Krippendorf's alpha: 0.30 for urgency) and as the performance of the model heavily depends on the initial labels it can greatly be impacted if the averaged ratings don't represent the true labels.

The low number of labelled instances (150 labels) add another major limitation relating to validation. The results stated for any performance metric on JAVA2015 dataset are validated on 150 labels that are available for this dataset using a 10-fold cross validation. Though using cross-

validation ensures that the instance used for validation is new and is not used during training but the low number of instances for validation fluctuates the results heavily even for a couple of misclassifications.

## 8.2 Future Work

This research study answered some useful and interesting research questions and has potential to answer a few more if the work is extended. The survey results could have been better in terms of agreement, had we given some prior training to the survey takers about our goals and objectives. Establishing a common rating scale for all survey takers could have effectively reduced the disagreements caused due to individual rating bias on Likert-like scale that was used in the survey. On a scale from 1-5 (where 1 means least urgency and 5 means most urgent) many survey takers considered 3 as somewhat urgent, some considered it neutral while there were some who even considered it low urgency. Such differences in individual perception regarding rating scale could have been greatly reduced with some prior training. In future, if I take such a survey again, I would like to do it with better training and instructions for the survey takers to get better agreement.

A research study on cross-domain classifier performance across datasets from MOOC and blended courses is an interesting study with useful implications. I was able to test models trained on MOOC datasets on blended course but not vice versa due to the lack of labelled dataset for blended courses. In my future work, I would like to label the blended course dataset using the generalization from the survey results and further study cross-domain classifier performance. Models with better performance for cross-domain classifications and between MOOCs and blended course could be extremely useful because of their scalability. A research study in this direction could unearth differences and similarities between MOOC and blended course dataset that I would like to explore and account for in my future classification models.

Another area of my research which has scope for future work is in identifying linguistic features for each of the factors stated under section 4.4.3 (survey results) for determining urgency of a post. Identifying linguistic features can help us construct custom features for each of these types of posts that can better perform than word frequency measures like Tf-idf alone. In my future work I would like to explore the possibility of using a multiclass classification-based approach on different factors that influence urgency to better improve classification of urgent posts.

CHAPTER

# 9

# CONCLUSION

This research study tried to establish a definition for urgency (as viewed by instructors/teaching assistants) in student discussion forums. The survey establishes some important factors that instructors use to decide the urgency of a post. A further study into these factors and modelling our learning algorithm to incorporate them could greatly improve performance of existing models and future models in identifying posts that need immediate attention. Prior research has revealed correlation between participant confusion and dropouts [16] and lack of responsiveness in MOOC forums could be a contributing factor in student dropouts [17] [18]. The study explored automated and reliable identification of urgent student queries on discussion forums which could greatly help instructors/TAs to prioritize their responses.

The semi-supervised models showed that it is possible to train a moderately reliable classification models on very limited training instances to identify urgent posts. These models can be combined with other learning models to improve performance as shown in this study. The research also explored cross-domain classification performance across MOOCs and blended course and showed promising results. As the results demonstrate, using custom word features and few metadata along with linguistic features from the posts can effectively be used to train models to perform reasonably good across domains for the purpose of identifying questions and non-questions. This opens doors for further investigation into cross-domain classification task which generally perform poorly for

multi-class and content-related classification but performed reasonably for question/non-question classification.

Instructors have emphasized the need for better navigation mechanism for MOOC discussion forums for effective learning [13]. This work has a potential to improve instructor's efficiency in monitoring large online discussion forums by automating identification of urgent posts. The goal is to provide instructors with real time identification of urgent posts for better responsiveness for posts that need immediate attention. Timely intervention can lead to less confusion and higher course completion rates. As novel online teaching methods develop, its challenges will also need to be met. Identifying urgent posts could find useful application in automated response systems (automatic TAs) on student discussion forums and this work could be a starting point for more targeted research moving forward in this direction.

# REFERENCES


[1] T. R. Liyanagunawardena, A. A. Adams and S. A. Williams, "MOOCs: A systematic study of the published literature 2008-2012," *International Review of Research in Open and Distance Learning,* vol. 14, pp. 202-227, 2013.

[2] S. D., "Monetization Over Massiveness: Breaking Down MOOCs by the Numbers in 2016," *EdSurge,* 2017.

[3] J. Kay, P. Reimann, E. Diebold and B. Kummerfeld, "Moocs: So many learners, so much potential," *IEEE Intelligent systems,* vol. 28, pp. 70-77, 2013.

[4] A. McAuley, B. Stewart, G. Siemens and D. Cormier, "The MOOC model for digital practice," 2010.

[5] D. Gasevic, G. Siemens and S. Dawson, "Preparing for the digital university: a review of the history and current state of distance, blended, and online learning.," *10.13140/RG.2.1.3515.8483. ,* 2015.

[6] E. G. Cohen, "Restructuring the classroom: Conditions for productive small groups.," *Review of educational research,* vol. 64, pp. 1-35, 1994.

[7] S. Palmer, D. Holt and S. Bray., "Does the discussion help? the impact of a formally assessed online discussion on final student results," *British Journal of Educational Technology,* vol. 39, pp. 847-858, 2008.

[8] L. M. Harasim, "On-line education: A new domain," *In R. Mason & A. Kaye (Eds.), Mindweave: Communication, computers and distance education, New York: Pergamon Press,* pp. 50-62, 1989.

[9] L. Rossi and O. Gnawali, "Language independent analysis and classification of discussion threads in Coursera MOOC forums.," *In Proceedings of the Information Reuse and Integration,* pp. 654-661, 2014.

[10] J. kim, "Influence of group size on students' participation in online discussion forums," *Computers and Education,* vol. 62, pp. 123-129, 2013.

[11] S. Chaturvedi, D. Goldwasser and H. & Daumé, "Predicting instructor's intervention in MOOC forums," *52nd annual meeting of the association for computational linguistics, ACL 2014-proceedings of the conference,* vol. 1, pp. 1501-1511, 2014.

[12] A. F. Wise, Y. Cui, W. Q. Jin and J. Vytasek, "Mining for gold: Identifying content-related MOOC discussion threads across domains through linguistic modeling.," *Internet and Higher Education,* 2017.

[13] F. Hollands and D. Tirthali, "MOOCs: Expectations and reality. Center for Benefit-Cost Studies of Education," *Teachers College, Columbia University,* 2014.

[14] V. Peters and J. Hewitt, "n investigation of student practices in asynchronous computer conferencing courses," *Computers in Education,* vol. 54, pp. 951-961, 2010.

[15] L. P. Dringus and T. Ellis, "Temporal transitions in participation flow in an asynchronous discussion forum," *Computers in Education,* vol. 54, pp. 340-349, 2010.

[16] D. Yang, M. Wen, I. Howley, R. Kraut and C. Rose, "Exploring the effect of confusion in discussion forums of massive open online courses.," *Proceedings of the second (2015) ACM conference on learning @ scale,* pp. 121-130, 2015.

[17] K. S. Hone and G. R. El Said, "Exploring the factors affecting MOOC retention: A survey study," *Computers and Education,* vol. 98, pp. 157-168, 2016.

[18] X. Wang, Y. D., M. Wen, K. Koedinger and C. P. Rosé, "Investigating how student's cognitive behavior in MOOC discussion forums affect learning gains," *Proceedings of the 8th international conference on educational data mining,* pp. 226-233, 2015.

[19] J. C. Richardson, A. Koehler, E. D. Besser, S. Caskurlu, J. Lim and C. M. Mueller, "Conceptualizing and investigating instructor presence in online learning environments," *International Review of Research in Open & Distance Learning,* vol. 16, pp. 256-297, 2015.

[20] M. Wen, D. Yang and C. P. Rosé, "Sentiment analysis in MOOC discussion Forums: What does it tell us?," *Proceedings of the 7th international conference on educational data mining,* pp. 130-137, 2014.

[21] J. S. Wong, B. Pursel, A. Divinsky and B. J. Jansen, "Analyzing MOOC discussion forum messages to identify cognitive learning information exchanges.," *Proceedings of the association for information science and technology,* vol. 52, pp. 1-10, 2015.

[22] Z. Marzouk, M. Rakovic and P. H. Winne., "Generating learning analytics to improve learners' metacognitive skills using nstudy trace data and the icap framework.," *LAK@ LAK,* pp. 11-16, 2016.

[23] M. T. Chi and R. Wylie., "The icap framework: Linking cognitive engagement to active learning outcomes," *Educational Psychologist,* vol. 49, pp. 219-243, 2014.

[24] M. Vellukunnel, P. Buffum, K. E. Boyer, J. Forbes, S. Heckman and K. Mayer-Patel, "Deconstructing the Discussion Forum: Student Questions and Computer Science Learning," *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education,* pp. 603-608, 2017.

[25] 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, 2017.

[26] M. K. Chandrasekaran, M.-Y. Kan, B. C. Y. Tan and K. Ragupathi, "Learning Instructor Intervention from {MOOC} Forums: Early Results," *CoRR*, vol. abs/1504.07206, 2015.

[27] M. Jenders, R. Krestel and F. Naumann, "Which Answer is Best?: Predicting Accepted Answers in MOOC Forums," *Proceedings of the 25th International Conference Companion on World Wide Web*, pp. 679-684, 2016.

[28] J. Kim and J.-H. Kang, "Towards Identifying Unresolved Discussions in Student Online Forums," *Applied Intelligence*, vol. 40, pp. 601-612, 2014.

[29] D. Yang, M. Wen and C. P. Ros'e, "Towards Identifying the Resolvability of Threads in MOOCs," *EMNLP*, 2014.

[30] O. Almatrafi, A. Johri and H. Rangwala, "H. Needle in a haystack: : Identifying learner posts that require urgent response in MOOC discussion forums.," *Computers & Education*, pp. 1-9, 2018.

[31] A. Agrawal, J. Venkatraman, S. Leonard and A. Paepcke, "YouEDU: Addressing confusion in MOOC discussion forums by recommending instructional video clips.," *Proceedings of the 8th international conference on educational data mining*, pp. 297-304, 2015.

[32] A. J. Viera and J. M. Garrett, "Understanding interobserver agreement: the kappa statistic.," *Family medicine*, pp. 360-3, 2005.

[33] K. K., "Estimating the reliability, systematic error, and random error of interval data," *Educational and Psychological Measurement*, vol. 30, no. 1, pp. 61-70, 1970.

[34] J. Ramos, "Using tf-idf to determine word relevance in document queries," *In Proceedings of the first instructional conference on machine learning*, Vols. 133-142, p. 242, 1999.

[35] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai and A. Kappas., "Sentiment strength detection in short informal text.," *Journal of the American Society for Information Science and Technology*, vol. 61, pp. 2544-2558, 2010.

[36] E. Stamatatos, G. Kokkinakis and N. Fakotakis, "Automatic text categorization in terms of genre and author," *Computer Linguistics*, vol. 26, pp. 471-495, 2000.

[37] B. Pang, L. Lee and S. Vaithyanathan, "Thumbs Up?: Sentiment Classification Using Machine Learning Techniques," *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing*, vol. 10, pp. 79-86, 2002.

[38] L. M. Manevitz and M. Yousef, "One-class svms for document classification," *Journal of Machine Learning Research*, vol. 2, p. 139–154, 2002.

[39] A. Dempster, N. Laird and D. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society, Series B. 39* , pp. 1-38, 1977.

[40] D. Zhou, O. Bousquet, T. N. Lal, J. Weston and B. Schölkopf, "Learning with local and global consistency," *Advances in Neural Information Processing Systems 16,* pp. 321-328, 2004.

[41] F.-R. Lin, L.-S. Hsieh and F.-T. Chuang, "Discovering genres of online discussion threads via text mining.," *Computers & Education,* pp. 481-495, 2009.

[42] K. J. Berry and P. W. M. Jr., "A generalization of cohen's kappa agreement measure to interval measurement and multiple raters," *Educational and Psychological Measurement,* vol. 48, pp. 921-933, 1998.

[43] K. Reusser and C. Pauli, "Co-constructivism in Educational Theory and Practice," *International Encyclopedia of the Social & Behavioral Sciences,* 2015.

[44] X. Zhu and Z. Ghahramani, "Learning from labeled and unlabeled data with label propagation," *School of Computer Science, Carnegie Mellon University,* 2002.