Assessing MOOCs Discussion Forums

Gonçalo Varela
goncalo.varela@ist.utl.pt

Instituto Superior Técnico, Lisboa, Portugal

October 2015

Abstract

With the recent popularity of Massive Open Online Courses (MOOCs), instructors of these courses now face the problem of having to choose from among the hundreds of posts from forums of their courses, who need more immediate response. In this thesis we use techniques from the fields of Sentiment Analysis and Natural Language Processing to classify / extract information from these posts, with the future goal of contributing to this choice. In a first study, we developed a classification model that allows us to identify whether a message belongs to an instructor or student. The best F-measure (F1) results (80.76 % for student category and 80.38 % for instructor category) were obtained using unigrams as features and having used a stemmer on the posts. The second study aimed at implementing a classifier that detects the polarity of a post based on the polarity of their terms, as defined in SenticNet lexicon. The best F-measure (F1) results were 74.6 % in the identification of positive terms beating the results of SentiStrength, and 38.3 % in the identification of negative terms. Finally, a third study identified the most frequently used expressions by instructors and students as well as the most common terms to the different and specific courses. We concluded that there are in fact specific expressions that could help in the identification of students and instructors as well as for the different courses.

Keywords: MOOCs, student, instructor, polarity, discussion forums, posts, sentiment.

1. Introduction

Massive Open Online Courses (MOOCs) are an in expansion web-based resource for e-learning that offer students the possibility of distance education. These courses are usually free and offer the opportunity of proper education to people, only requiring a computer and an internet connection. There are numerous companies that provide MOOCs. Amongst the most notorious: Coursera.org[1] and edX[2]. These courses, like traditional classes, comprise several aspects of presential learning where students can attend and participate in different classes.

One of the most important features in MOOCs are the discussion forums. These forums are the only way that students have to interact with the instructors, allowing them to submit their work and share their knowledge and doubts, not only with instructors but also with others students, forming what we can call an online community.

Due to the large amount of participants in these courses, these forums tend to have large amounts of posts and threads making the instructors' job very difficult in terms of data analysis. Thus, we will work on some tasks that could assist the instructors, such as the detection of written students' emotional expressions.

In this work we are going to analyze discussion forums of Coursera courses to determine, based on the posts of instructors and students in a training set, which features offer better results in the classification of posts' authors referring to instructors and students. These forums were used as corpora during the development of this thesis and were kindly provided by Lorenzo Rossi, [14]. The purpose of this study, is to establish, based on our dataset that already include this type of information, a starting point and useful knowledge for helping future works where this kind of information may be lacking. Then, we will propose an approach to evaluate the polarities of these posts, making it possible to instructors identify which posts are most likely to contain doubts or dissatisfaction by students. Also, we will identify in all courses which students' expressions are the most commonly used.

In this thesis we discuss topics of different study fields, such as **Natural Language Processing (NLP)** to address text processing techniques to assess the written posts of students, and **Opinion Mining/Sentiment Analysis** to determine the emotion polarities and the affect these words and

[1] https://www.coursera.org/
[2] https://www.edx.org/

expressions infer. These assignments will have the assistance of useful resources such lexicons that will provide the polarities and scores for the classification process.

This document is organized as follows: In Section 2 we present the dataset we will use and the related work. In Section 3 we specify the methodology for training a classifier to distinguish the post authors in a dataset. The task of developing a methodology for polarity assessment in students posts are explained in Section 4. In Section 5 a study of the most common expressions used in MOOCs is made. Finally, in Chapter 6, we present a conclusion and a starting point for future work.

2. Related Word Used

2.1. Coursera Discussion Forums

For this work a dataset containing threads from forums of 60 different courses of Coursera [3] was thoughtfully facilitated by Lorenzo Rossi who worked on it for a paper, Rossi and Gnawali [14]. For his paper, the author studied these forums and captured data that gives insight about the basic data of the courses (name, id, duration, number of users and threads, *etc*); threads and subforums (course and thread ids, name of the forums, *etc*); and data about the posts (posts, threads and courses ids, the author and their type, number of votes, *etc*).

2.2. EmotiWorld

EmotiWorld[4] is a lexicon conceived only with emoticons. Usually each emoticon represents an emotion state and consist in using several characters and symbols, like punctuation, to draw in most cases, a representation of a facial expression, like a smile (:)) or a grin (:D). EmotiWorld contains a database with emoticons to express and represent states such as happiness, sadness, surprise, confusion among others. In discussions forums these emoticons are largely used and to address them the EmotiWorld could be useful.

2.3. SentiWordNet

In their work Esuli and Sebastiani [5] describe another resource for opinion mining, named SentiWordNet. SentiWordNet also makes use of the WordNet synsets and allocate to each of their terms, one of three possible labels: objective, positive or negative. Also it quantifies the association of these labels with the synsets through a numerical score ranged between 0.0 and 1.0.

2.4. SenticNet

SenticNet developed by Cambria et al. [1] is a public available opinion mining resource containing the most common used concepts associated with polarities, and their respective scores regarding the strength of its positivity and negativity. In contrast with SentiWordNet it does not contain words with neutral polarity being them eliminated when their scores are neutral. Also, it takes into account not only polarities at a syntactical level but also in a semantic level being able to attribute polarities at concepts such 'accomplish goal' or 'bad feeling'.

2.5. AFINN

The AFINN, Nielsen [11], is an affective lexicon labeled by Finn Arup Nielsen containing a list of 2477 English words and phrases along with its valence from a range between -5 and 5, with the particularity of including obscene words.

This lexicon has the particularity of taking into consideration Internet slang acronyms, like "LOL", "WTF" or "LMAO" which are also scored regarding its sentiment strength and can be useful when working with short informal texts like microblogs or forums.

This lexicon started with a version of 1468 different words (AFINN-96) and then updated to a version containing 2477 words (AFINN-111).

2.6. SentiStrength

SentiStrength (Thelwall et al. [16]) is an algorithm developed for sentiment detection focused in user behavior. It extracts the sentiment strength from English texts contemplating the spelling styles of cyberspace. The SentiStrength predicts the sentiment strength of texts in a range of [-5,-1] for negative emotions and [1,5] for the positive ones, differing from other different sentiment detection algorithms by mixing both positive and negative emotions. In this way a written text can possess both positive and negative scores which according to a psychological study it is how we process emotions.

This algorithm started to be developed exploring a corpus of from 2600 MySpace comments and resorts to a machine learning approach to enhance the sentiment weights of the terms.

2.7. The Difficulties of Sentiment Classification and Possible Solutions

This study addressed the potential problems that a general purpose tool could face and a set of alternatives were discussed. One of the problems is the erroneous attribution of polarity to emotional expressions. There are several situations where this problem can be found, such as negation, common sense, points of view, *etc*. For this research, the datasets from SemEval-07 affective task with 1000 utterances and SemEval-13, containing 7500 utterances, were employed.

Denis et al. [3] revised some works and approaches to respond the complexity of these prob-

[3] https://www.coursera.org/
[4] http://en.emotiworld.com/

lems, such as: the use of machine learning with an unsupervised approach which used Pointwise Mutual Information (PMI) to assess the differences in the valence of words in reviews; the use of annotations to label reviews to train classifiers such as Support Vector Machine (SVM); and the study of Conditional Random Fields (CRF) (Nakagawa et al. [10]) and autoencoders (learning networks that transform efficiently inputs into outputs with the least possible amount of distortion Rumelhart et al. [15]), to assess sentimental analysis.

Also, this general purpose tool took into consideration three other different problems: domain dependence, interoperability and multilinguality.

The first problem refers to the use of supervised machine learning and its dependence to the available training sets. To solve this problem, the most appropriate approaches are the semi-supervised and unsupervised machine learning, and hybrid methods. The interoperability problem is related with the lack of consensus in emotional representation. W3C[5] proposed a recommendation for these representations, the EmotionML. The problem of the multilinguality is that most of the works and developments are made only for the English language. To assess this problem different researches have been made using, for example, translated lexicons or the use of training classifiers in translated corpora.

2.8. Sentiment Classification Algorithm

In SemEval-2013, a team (NRC-Canada) formed by Mohammad et al. [9] developed two classifiers to detect emotions on both message-level and term-level tasks. Two SVM for sentiment detection (positive, negative or neutral) were created, one for tweets and Short Message Service (SMS) and other for terms within a message. For this research, two lexicons were developed as well as semantic and sentimental features. The results of the algorithms were good, being ranked in first on both classifiers.

For this competition two datasets were provided by the organization, one for tweets with labels regarding the sentiment and divided in sets for training, development and testing, and one for SMS where the total of messages were used for testing (no training or developments sets used).

Sentiment lexicons were created using other established lexicons like NRC Emotion Lexicon (Mohammad and Turney [7], Mohammad and Yang [8]), Multi-Perspective Question Answering (MPQA) Lexicon (Wilson et al. [17]) and Bing Liu Lexicon (Hu and Liu [6]). The add-ons made to these new lexicon included hashtag sentiment polarity detection, being the hashtags from tweets extracted and analyzed. The previous mentioned PMI was used to calculate the score of a term regarding their association with positive (or negative) sentiments. Also, different pairs of unigrams, bi-grams, and a combination of both were generated and some punctuation removed. The second lexicon was created using the same methodology but dedicated to sentiment emoticons instead of hashtags.

SVM was applied for sentiment detection in messages. The tweets were normalized, tokenized and Part of Speech (POS) tagged. Each tweet had a feature vector containing the features: word n-grams and character n-grams, the number of words written all in caps, the number of ocurrences of each POS tag, the number of hashtags, lexicons, punctuation, emoticons, elongated words, clusters and negations.

SVM was also used with a linear kernel for the automatic sentiment detection of terms in a message, and the features applied in this classification were: word and character n-grams, elongated words and punctuation if were present, emoticons, upper case, stopwords, lenghts, negation, position of the term (beggining, end, or another position), sentiment lexicons, term splitting, and others (if a term contained an user name or an URL).

For evaluation purposes, the classifiers were applied to training, development and testing sets and measured by F-score, obtaining the results of 69.02 and 88.93 in the message-level and term-level tasks, respectively.

3. Distinguishing Students' from Instructors' Posts in Education Forums

We adopted the dataset specified in Section 1.2 as the primary source of information containing all the posts used in the implementation and development of our tasks. We conducted a series of preprocessing tasks to format these data onto a set of posts proper to use, where in each line of a text file we have – AuthorType: Post.

3.1. Architecture

For this task, the goal was to determine, based on the dataset containing the posts and respective authors, the best way to classify the data by identifying the author (student or instructor) of a post through its terms. After the pre-processed phase, the data was classified using a classifier along with a set of features.

In order to reach the best outcome possible and optimize the capture of expressions that would indicate if a post belong to an instructor or a student, the dataset was subjected to other different approaches, such as the removal of stopwords and the use of stems, to assess which of these experiences could offer better results on identifying authored expressions.

The introduction of a stemmer was employed in the original dataset to allow the capture of more

[5] http://www.w3.org/

terms and expressions reducing all terms to its root form and thus expand the number of equal terms and expressions that before were written in different verbal forms.

All posts have their category recognized by the classifier which then, according with a set of features applied on the posts, determine to which class the post belong according to its terms and the resulting measures from this classification.

3.2. Development

After cleaning the data, an excerpt of the dataset was randomly selected as training set containing 10,000 posts, 5,000 from each author type.

A SVM Machine Learning classifier with a linear kernel type was chosen. The choice of this classifier was made after the study of some papers (*i.e.* section 2.8) related with this area of specialization, concluding that this type of classifier was majority considered and used in those works.

This type of classifier was implemented in the TalKit (Dias [4]), which we used and modified in order to satisfy our requirements. In the TalKit classifier, in order to determine the class or category of a post between two possible classes âĂŞ *Instructor* or *Student*, a modification was performed consisting in the conversion of the format of how the corpus would be represented, transforming it to the format: "category : post".

For evaluation purposes, the TalKit classifier allows to evaluate our dataset using a k-fold cross-validation technique, and in addition to the accuracy measure that it had already implemented and was able to calculate, we added the possibility to estimate the precision, recall and F-measure scores for a more complete evaluation setup in our project. These measures are usually used in Information Retrieval and Text Classification as referred by Rijsbergen [13]. Furthermore, a status of how the post were being classified was added, showing if a post was correctly being classified or not.

With the classifier and evaluation code appropriately modified, some classification features that were already implemented were used, consisting in different ngrams combinations and sizes, in order to verify which offered better results. The following set of features was used: Unigrams (U), Bigrams (B), Trigrams (T) (Figure 3.20); and all combinations of those, including: Unigrams+Bigrams (U+B), Unigrams+Trigrams (U+T), Bigrams+Trigrams (B+T) and finally Unigrams+Bigrams+Trigrams (U+B+T).

For the evaluation of the models, both 10-fold cross-validation and the added measures precision, recall, accuracy and F-measure were used. These measures were calculated using as expected category the Instructors and Students posts independently. These experiments were repeated posteriorly for different training sets: one without stopwords (words that are normally used as connectors and are generally the most common words in a language) and other only using the stems of the posts.

3.3. Results

The dataset and both experiments of removing stopwords and employing a stemmer to the original dataset were evaluated and ran with the classifier to estimate which features would be best employed to have the better accuracy and F-measure values. As for the F-measure, the values of precision and recall were calculated, using both categories Instructor and Student as the expecting value.

Table 1: Results of the classification on dataset considering Instructor as expected category.

Ngrams	Precision	Recall	F-Measure
U	79.88	78.51	**79.16**
B	75.29	69.11	72.04
T	73.64	57.53	64.55
U + B	80.56	77.69	79.09
U + T	80.18	77.57	78.84
B + T	71.25	74.92	73.28
U + B + T	80.16	77.66	78.88

An accuracy of 75.94% was reached (Table 1), and the features that showed better F-Measure value was the feature unigrams with the highest score with 79.16%.

Table 2: Results of the classification on dataset considering Student as expected category.

Ngrams	Precision	Recall	F-Measure
U	79.54	80.39	**79.94**
B	72.02	75.48	73.65
T	65.46	79.36	71.73
U + B	78	78.7	79.2
U + T	78.51	81.06	79.75
B + T	74.5	71.83	73.12
U + B + T	77.89	80.22	79.03

The feature who showed a higher value in F-Measure (Table 2) was the unigrams with a score of 79.94%, and the accuracy of this classification was 76.04%.

In the experience where the classifier received as corpora the original dataset without their stopwords, and when the category Instructor is evaluated (Table 3), the feature that provided the highest F-measures results with a value of 79.51% was the combination of features unigrams and bigrams. When the evaluation category was Student (Table 4), the highest score of F-Measure belong to the feature where it was combined unigrams and bigrams

Table 3: Results of the classification on dataset without stopwords considering Instructor as expected category.

Ngrams	Precision	Recall	F-Measure
U	79.49	78.28	78.87
B	75.03	66.99	70.76
T	81.66	35.2	49.18
U + B	81	78.12	**79.51**
U + T	80.63	77.46	78.99
B + T	75.13	67.17	70.79
U + B + T	80.66	77.09	78.82

Table 4: Results of the classification on dataset without stopwords considering Student as expected category.

Ngrams	Precision	Recall	F-Measure
U	78.78	80.16	79.44
B	70.43	78.39	74.17
T	59.01	91.8	71.83
U + B	78.63	81.65	**80.1**
U + T	78.34	81.06	79.65
B + T	69.84	78.01	73.68
U + B + T	78.17	81.02	79.54

with a value of 80.1%. Regarding the accuracies of these classifications, when the evaluation category was set to Instructor, it scored 75.15%, and a value of 75.2% when its category was Student.

Table 5: Results of the classification on dataset only using stems, with Instructor as expected category.

Ngrams	Precision	Recall	F-Measure
U	81.11	80.46	**80.76**
B	74.91	72.25	73.5
T	73.71	60.23	66.27
U + B	79.81	78.35	79.05
U + T	80.53	77.95	79.2
B + T	73.18	75.5	74.3
U + B + T	80.36	78.5	79.26

Table 6: Results of the classification on dataset only using stems with Student as expected category.

Ngrams	Precision	Recall	F-Measure
U	80.24	80.55	**80.38**
B	73.13	76.57	74.56
T	66.33	78.55	71.88
U + B	78.65	81.04	79.82
U + T	79.05	81.06	80
B + T	74.58	71.77	73.13
U + B + T	78.65	80.65	79.62

Regarding the approach where we applied a stemmer to all terms of the posts, and when evaluating the category Instructor (Table 5), it scored an accuracy of 76.63%. The resulting values of the use of the classifier demonstrate that when applied the unigrams feature it provided a better f-measure result with a value of 80.76%. For the same dataset but a classification where the evaluated category was Student (Table 6), the accuracy of the classification was 76.55%, and the feature that showed higher score were also the unigrams with an f-measure value of 80.38%.

4. Polarity assessment in students posts

4.1. Architecture

The objective with the development of this task was to offer instructors the possibility of determining in an easier way, the polarity and valence of the students' posts in order to determine which need to be aided and which show satisfaction in their posts.

In order to detect the sentiment expressed in students' posts, the lexicon SenticNet was exploited, and the information containing the terms and expressions was extracted along with the respective polarities, to be crossed with the same terms and expressions found in the posts. These posts were subjected to a set of experiments to assess if it would be possible to capture more terms and expressions due to some verbal conjugations or some expressions where its stopwords were ignored.

The introduction of a stemmer in the original dataset as an experiment was considered due to the fact that a large number of terms and expressions raised by students were not always in the same verb form and therefore wasn't always contemplated by the lexicon even though they have the same sentimental value. The application of a stemmer in both the lexicon and the posts allows to normalize all terms and expressions to be possible to take the greatest number of occurrences of these and thus obtain a more accurate classification.

The final polarity of each post were then calculated through the arithmetic mean of its terms' polarities, which was considered more suitable for this classification instead of summing all terms' polarities, as it takes into consideration the number of terms of the post present in the lexicon and thus granting that smaller posts with fewer terms could be evenly weighted.

Finally, the posts are presented to the instructor with the possibility of showing only the posts whose polarity value is lower than zero, due to be considered as those with highest priority.

In addition to this classifier, and based in the same process, another functionality was created to be deployed in other projects, that instead of offering a set of posts, it allows the classification of a single sentence along with the lexicon that we want to consider and apply to classify it. This functional-

ity was developed to offer other projects of Question and Answer (Q&A) a resource to evaluate submitted questions with their sentiment polarity.

4.2. Development and Evaluation Setup

For the development of this task, we studied the SenticNet lexicon to determine what would be necessary and useful for the implementation of our task. In SenticNet all concepts have associated a set of values referring the degree of pleasantness, attention, sensitivity, aptitude, polarity and also synsets (set of synonyms) with all their related terms also present in the lexicon.

Taking this into account, the next step was to format the SenticNet lexicon into a text file with its concepts along with their respective polarities.

The classifier begun to be developed by creating the structures for our data. The SenticNet concepts were then saved in a hash map containing the concepts and its values, as well as the posts that were also saved in a hash map alongside with an id to identify them.

After the posts and concepts properly added to the structures, lists of terms began to be populated. It started reading each line of the posts file at a time and for each line, all the terminations "n't" for the word "not" were replaced in order to capture and invert the polarity of the concepts that follows it. The category of each post – *Student:* or *Instructor:* was removed as well as the punctuation marks, ".", ";" and ",". Finally, the "?" and "!" symbols was also separated from the words by a whitespace character to be possible the inclusion of these words on our search. Also, all words were posteriorly formatted to lower case so they could be treated and classified equally.

In order to improve the accuracy in weighting some expressions not considered in the lexicon, a set of adverbs was used. If the word immediately before the term is one of these adverbs, the polarity value of the term would be emphasized or depreciated depending on the adverb found. These adverbs were treated like the already mentioned word *not*, but instead of reversing the polarity of the term, its polarity was increased or diminished. Among these adverbs, the positive ones were *quite*, *very*, *more* and *several* while the negative adverbs were the words *few* and *less*.

With the sentences formatted, each word was crossed with all concepts of the lexicon taking into consideration previous and posterior words and thereby allow the capture of expressions. For example, both terms "get angry" and "angry" are present in the lexicon, so we needed to consider that when "get" appeared before the term "angry" the concept to be selected would be "get angry" instead of "angry".

After the matching, to prevent the classifier to consider both the term and expression that contains the same term, for example, that both of the terms "angry" and "get angry" were considered and added to the post, was used a function to remove the words that were a substring of another, and thus the concept "angry" was not added to the list.

Upon all the populated structures, for each set of terms of a post, their polarities were summed in order to calculate the polarity of the post and fractioned with the number of terms to achieve an arithmetic mean.

For the terms that started with the word *not* its value was multiplied with -1 to invert its value. The same treatment has been made for the adverbs mentioned above. The terms that contained one of the positive adverbs were multiplied by 1.25, while the negative adverbs (*few* and *less*) by 0.75.

To complete the task, we implemented a notification method, to only consider the negative posts. The final results are shown in a format – Polarity: <polarity> -> Post: <post>.

Finally, an evaluation class to estimate the scores of precision, recall, accuracy and f-measure of our classifier when ranking the posts using a specific lexicon was built.

In order for our classifier to work with other sets of posts and languages, we had the attention of giving files as parameters to our classifier. These files contain all of the exterior information, such as the lexicons and the posts file, to be received as input so our classifier could be used and extended to work with a widely set of corpora.

4.3. Evaluation Setup

To create a reference to classify this project, a set of data containing 100 students' posts was annotated, which was randomly selected from the completed dataset.

These posts were then classified by 2 annotators, which categorized them by being positive, neutral or negative.

After rating these posts, a concordance of 56.1% between both annotators was reached using an inter-annotating agreement and determining as measure the Cohen's kappa values Cohen [2]. This value can be explained by the high level of disagreement in neutral posts due to their ambiguity.

The sentence polarity dataset v1.0 of Pang and Lee [12] was also used as reference to test our classifier. In this dataset, 5,331 sentences are classified as being positive or negative.

Other reference we used, to evaluate our classifier rating texts out of the MOOCs scope, was the SpeDial results on a set of movie texts. The SpeDial is a Spoken Dialogue System that have the ability to perform affective modeling of spoken dialogue.

It is a dataset containing 3826 rated texts that we used as reference, and compared with the results achieved by our classifier.

Our classifier was also used and evaluated using a different lexicon, the AFINN lexicon (section 2.5), and the SentiWordNet lexicon (section 2.3) was studied to infer how it could be used and applied.

In order to determine the best classifier in measuring the polarity strengths of posts, the performance of this classifier was compared with the sentiment analysis tool SentiStrength (Section 2.6) to assess which one would offer better results.

4.4. Results

Table 7: Results of the classification on different experiences when expecting a positive result. SW – stopwords, St – stems

Lexicon	Pre	Rec	Acc	F-M
SenticNet (SN)	63.49	76.92	59	69.57
SN w/o SW	64.06	78.85	60	70.69
SN w/St	63.51	90.38	**62**	**74.6**
AFINN (AF)	72.72	46.15	51	56.47
AF w/o SW	72.72	46.15	51	56.47
AF w/o St	74.42	61.54	57	67.37

Table 8: Results of the classification on different experiences when expecting a negative result. SW – stopwords, St – stems

Ngrams	Pre	Rec	Acc	F-M
SN	40.91	34.62	59	37.5
SN w/o SW	42.86	34.62	60	38.3
SN w/St	44.44	30.77	**62**	36.36
AF	52.63	38.46	51	44.44
AF w/o SW	55.56	38.46	51	**45.45**
AF w/St	55.56	38.46	57	**45.45**

The results of this evaluation showed that our classifier was more accurate when employing the lexicon SenticNet with stems with an accuracy score of 62%, and when the expected result was the positive one it also showed the best score in calculating the f-measure with a value of 74.6% (Table 7). Regarding the estimation of the f-measure when the negative result was expected (Table 8), the lexicon AFINN without stopwords and with stems had the same highest result of 45.45%.

According to the SentiStrength algorithm, the experiment that offered the best results when comparing their evaluation with the data annotated was the approach where we removed the stopwords from the original dataset reaching an accuracy of 46% and an f-measure of 53.01% when expecting a positive result (Table 9) and 40% when evaluating for a negative result (Table 10).

Table 9: Results of the SentiStrength algorithm on different experiences when expecting a positive result. SW – stopwords, St – stems

Ngrams	Pre	Rec	Acc	F-M
SStrength (SS)	70	40.38	45	51.22
SS w/o SW	70.97	42.31	**46**	**53.01**
SS w/St	62.96	32.69	42	43.04

Table 10: Results of the SentiStrength algorithm on different experiences when expecting a negative result. SW – stopwords, St – stems

Ngrams	Pre	Rec	Acc	F-M
SS	45	34.62	45	39.13
SS w/o SW	47.37	34.62	**46**	**40**
SS w/St	42.11	30.77	42	35.56

After evaluating our classifier and comparing it with other sentiment detection algorithm, our classifier was subjected to other dataset out of the domain of MOOCs to assess their performance. For this last experiment, we recurred to a set of texts used and already evaluated and scored by the Spedial project and compared their results with the results achieved with our classification (Table 11).

Table 11: Results of our classifier when compared with the Spedial project scores.

Expected Result	Pre	Rec	Acc	F-M
Positive	46.16	65.1	45.27	54.02
Negative	82.42	29.93	45.27	43.92

This experiment showed that our classifier had an accuracy of 45.27% when estimating the sentiment strength of the texts used in Spedial, and that their performance was higher when evaluating the positive results showing a score of 54.02%

Regarding the results of our classifier when evaluating the sentences from the Pang&Lee dataset (Table 12), we concluded that our classifier perform better when classifying positive texts, showing an accuracy of 84.51% and f-measure score of 91.61%, than when classifying negative texts, which scored an accuracy of 26.82% and f-measure of 42.3%.

Table 12: Results of our classifier when compared with the Pang&Lee sentences dataset scores.

Expected Result	Pre	Rec	Acc	F-M
Positive	100.0	84.52	84.51	91.61
Negative	100.0	26.82	26.82	42.3

5. Most common expressions used in MOOCs

The kfNgrams [6] is a free-software for linguistic research that through a text file given as input determines the most frequent n-grams.

In Table 13 we present the expressions regarding the kfNgrams results, which allowed us to assess which expressions were used with higher frequency by both students and instructors, being identified by their author type. For this study, we used the complete pre-processed dataset containing all posts from all courses, and divided it in two parts, one containing only the posts from students and other only with instructors' posts.

For this study, we used 5-grams to find the most common expressions due to the presence of stopwords in these sentences, which although does not directly provide any utility, when used as connectors can interfere with the construction of different expressions, and their absence could interfere with the intention of some expressions, on the other hand the use of less n-grams could not be enough to capture relevant expressions.

Table 13: The most common expressions belonging to instructors and students when searching with 5-grams.

Instructors	"you should be able" "at the end of the" "take a look at the" "you will be able to" "at the top of the" "let us know if you" "thank you for your feedback"
Students	"at the end of the" "this is my first course(coursera)" "hello(hi) everyone my name is" "it seems to me that" "thank you very (so) much for" "there are a lot of" "nice to meet you all"

The analysis of the results allowed to conclude that most of the expressions used by instructors are those which they give indications about something that was asked ("at the end of the", "at the top of the", "take a look at the"), showing their ability to answer students' questions and doubts. Also, expressions of encouragement ("you should be able to") and promotion of participation ("let us know if you") were found, suggesting their active presence in the forums and their will to address any students doubts, as well as sentences expressing gratitude about something that was suggested or criticized in order to improve it ("thank you for your feedback").

[6] http://kwicfinder.com/kfNgram/kfNgramHelp.html

Regarding the students posts, we found that the most common used expressions consisted in posts where students present themselves or state their inexperience taking online courses, at least in Coursera, ("hello(hi) everyone my name is", "nice to meet you all", "this is my first course(coursera)"), in expressions where they are grateful for the assistance provided ("thank you very(so) much for"), and expressions where they give their personal opinion about an occurrence ("it seems to me that"). Other expressions were often found, suggesting their aptitude for helping other students, giving them indications as those given by the instructors ("at the end of the", "there are a lot of").

Other highly used expressions were found, but due to their presence in only a short sequence of posts, showing that these expressions were largely used in just one specific course, such as "brought out the best in" which are presented in a course of leadership or the sentence "use this thread to ask help and clarification about Question X, for your convenience here's the text of the question:" which is a pre-made sentence made by the instructors of the course "useful genetics".

These expressions, led us to study and discover the vocabulary used in the courses to determine the most common and specific terms for each available course in our dataset. Unlike the features used in the previous table where we made our search using 5-grams to find expressions, for this experience we will only use unigrams due to the fact that we only want the most common and specific terms and we will also exclude stopwords and words that we find transversal to all courses. An excerpt of these results are shown in Table 14.

6. Conclusions and Future Work

MOOCs are a recent and currently in expansion learning tool that allow individuals to enroll in online courses and receive proper education about several and different study domains. Due to its growth, the forums of these courses usually have thousands of posts making it difficult for instructors to assess students and evaluate them consistently.

To answer the large quantity of posts these courses usually have, this thesis proposed a model of classification and a sentiment polarity classifier in order to give instructors useful information regarding the posts. A study was also made, to determine the most frequent and common expressions in students' and instructors' posts, and in specific areas of learning.

The model of classification aimed at the automatic identification of the authors of the posts of the forums courses. For this task, we used the classifier developed in TalKit alongside a set of features to classify each post of our dataset. For this classification process the features used were the ngrams,

Table 14: The most common expressions belonging to instructors and students when searching with 5-grams.

Courses	Frequent terms
Asset Pricing	“price" “risk" “asset" “market" “value"
Climate Literacy	“climate" “change" “people" “energy" “carbon"
Designing Cities	“city(ies)" “urban" “design" “planning" “maps"
Game Theory	“player" “strategy" “game" “payoff" “equilibrium"
History of Rock	“music" “rock" “Beatles" “song(s)" “love"

namely unigrams, bigrams, trigrams and combinations between them. Also, these features were extracted from different experiments that were made to the dataset, one where the stopwords of each post were removed, and other where a stemmer was employed so it would only contain stem words. After this task was evaluated, we concluded that to obtain the best results at identifying the authors of the posts, the features that should be used are unigrams or combinations of them with other type of ngrams. Also, the reduction of all words to stems from the dataset slightly improved the classifier results.

A sentiment detection classifier was also implemented that alongside with available lexicons, allows to estimate the sentiment strength of the posts written by students in our dataset and how its results would be influenced by the removal of stopwords and the use of stems. In this classifier, we determine the polarity and valence of the students posts according to their sentiment terms present in the lexicon.

Also, this classifier was evaluated using different lexicons, and sets of data out of the context of MOOCs (Spedial and the Pang&Lee dataset) and its results compared with other sentiment detection algorithm, the SentiStrength.

With this task, we concluded that our classifier showed better results when evaluating positive posts, and with the use of stems, which can be explained by the English language does not use its verbs and terms in their root form when establishing a conversation and that the negative sentiments written by the students are usually expressed through expressions difficult to capture in automatic classification, such as irony or sarcasm or other difficult of capture expressions.

To improve the results of our methodologies, future work can be conducted in order to improve our classifier estimating the valence and polarity of the posts. One of these tasks could be the adoption of POS tagging to allow the introductions of other type of lexicons, namely the SentiWordNet (Section 2.1.8) which differentiates the polarity strengths of the terms according its POS.

Other future work that can be done is an extensive process of annotation, that would allow to capture difficult expressions to answer in an automatic form, such as sarcasm, irony, or even frustration and boredom. These expressions are frequently used when expressing negative emotions and a method to identify them would greatly improve this task.

Also, punctuation marks can be useful when classifying students' posts. The use of exclamation mark can be used to increment the strength of a sentence as the interrogation mark used to discover questions from students.

Regarding the method used to calculate the sentiment polarity of a post, other measures such as the consideration of the dominant emotions of the posts should be taken into account.

References

[1] E. Cambria, R. Speer, C. Havasi, and A. Hussain. Senticnet: A publicly available semantic resource for opinion mining, 2010. URL https://www.aaai.org/ocs/index.php/FSS/FSS10/paper/view/2216.

[2] J. Cohen. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46, Apr. 1960. doi: 10.1177/001316446002000104. URL http://dx.doi.org/10.1177/001316446002000104.

[3] A. Denis, S. Cruz-Lara, and N. Bellalem. General purpose textual sentiment analysis and emotion detection tools. *CoRR*, abs/1309.2853, 2013. URL http://arxiv.org/abs/1309.2853.

[4] C. Dias. Talkit - desenvolvimento de um sistema de diÃąlogo para portuguÃłs, 2015.

[5] A. Esuli and F. Sebastiani. Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation (LRECâĂŹ06*, pages 417–422, 2006.

[6] M. Hu and B. Liu. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 168–177, New York, NY, USA, 2004. ACM. ISBN 1-58113-888-1. doi: 10.1145/1014052.1014073. URL http://doi.acm.org/10.1145/1014052.1014073.

[7] S. M. Mohammad and P. D. Turney. Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, CAAGET '10, pages 26–34, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1860631.1860635.

[8] S. M. Mohammad and T. W. Yang. Tracking sentiment in mail: How genders differ on emotional axes. In *Proceedings of the 2Nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, WASSA '11, pages 70–79, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 9781937284060. URL http://dl.acm.org/citation.cfm?id=2107653.2107662.

[9] S. M. Mohammad, S. Kiritchenko, and X. Zhu. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. *CoRR*, abs/1308.6242, 2013. URL http://dblp.uni-trier.de/db/journals/corr/corr1308.html.

[10] T. Nakagawa, K. Inui, and S. Kurohashi. Dependency tree-based sentiment classification using crfs with hidden variables. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 786–794, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. ISBN 1-932432-65-5. URL http://dl.acm.org/citation.cfm?id=1857999.1858119.

[11] F. Å. Nielsen. Afinn, mar 2011. URL http://www2.imm.dtu.dk/pubdb/p.php?6010.

[12] B. Pang and L. Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the ACL*, 2005.

[13] C. J. V. Rijsbergen. *Information Retrieval*. Butterworth-Heinemann, Newton, MA, USA, 2nd edition, 1979. ISBN 0408709294.

[14] L. A. Rossi and O. Gnawali. Language Independent Analysis and Classification of Discussion Threads in Coursera MOOC Forums. In *Proceedings of the IEEE International Conference on Information Reuse and Integration (IRI2014)*, Aug. 2014.

[15] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Neurocomputing: Foundations of research. chapter Learning Internal Representations by Error Propagation, pages 673–695. MIT Press, Cambridge, MA, USA, 1988. ISBN 0-262-01097-6. URL http://dl.acm.org/citation.cfm?id=65669.104449.

[16] M. Thelwall, K. Buckley, G. Paltoglou, and D. Cai. Sentiment strength detection in short informal text, 2010.

[17] T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 347–354, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics. doi: 10.3115/1220575.1220619. URL http://dx.doi.org/10.3115/1220575.1220619.