ENUG 2021 Day 3: Using Natural Language Processing and Machine Learning Techniques to Analyze Library Chat Service Data: A Pilot Project (Yongming Wang)

Thursday, 10/28 4:00 PM

Transcript generated by Zoom Live Transcription

00:00:00.000 --> 00:00:13.000

There's.

00:00:13.000 --> 00:00:19.000

Hello everyone, my name pronounced Yeoman one and assistance librarian at the economy of New Jersey.

00:00:19.000 --> 00:00:40.000

I'm the assistant librarian at the economy of New Jersey. Today my topic is using natural language processing and machine learning techniques to analyze the library virtual reference data.

00:00:40.000 --> 00:00:43.000

This is the outline.

00:00:43.000 --> 00:00:47.000

I'm going to talk about three parts.

00:00:47.000 --> 00:00:49.000

The project description.

00:00:49.000 --> 00:00:56.000

The foundations and the project itself.

00:00:56.000 --> 00:01:16.000

I think my problem is the only one, which is is not related to any x Nibiru product, so I feel like I'm a outliers

00:01:16.000 --> 00:01:20.000

project description, first of all is the data.

00:01:20.000 --> 00:01:34.000

The data is the transcripts of eight years chatter reference and transaction, you are you my library from 2014 to 2021.

00:01:34.000 --> 00:01:47.000

So the research question is, there are many types of questions asked the by chatter younger part of the big two categories are the reference questions, and the non reference questions.

00:01:47.000 --> 00:02:03.000

So the question is, is it possible to build a smart field or the other well differentiated a reference questions from non reference questions

00:02:03.000 --> 00:02:20.000

they matter and it's gold. I want to use the data available to build a classification model by using an LG and ml techniques. This model might be used in the future to predict the category.

00:02:20.000 --> 00:02:35.000

like a raft question or wrong last question of new questions to ask it up by library patrons. So probably, it will improve the efficiency and effectiveness of the Virtual Reference service of the library.

00:02:35.000 --> 00:02:43.000

The language I use the Python, and as a development tool I use is the Jupiter notebook.

00:02:43.000 --> 00:02:54.000

So before I talk about the next next section. A little background off of this project.

00:02:54.000 --> 00:03:05.000

The artificial intelligence and machine learning, nowadays are everywhere. I don't think I needed to talk more about not.

00:03:05.000 --> 00:03:29.000

The my knowledge and the skill. I used to be a computer programmer for a few years. I did a full time job on developing in cooperation and a few years ago, I took online, six months, online course to gain the certification in data science.

00:03:29.000 --> 00:03:33.000

That is like three or four years ago.

00:03:33.000 --> 00:03:36.000

So, naturally.

00:03:36.000 --> 00:03:42.000

I asked myself how to apply all this to the library setting.

00:03:42.000 --> 00:03:52.000

That's the whole the idea of this project, come out.

00:03:52.000 --> 00:04:10.000

So the project is started in 2019 it, the cave. Preliminary talk at a conference to two years ago Atlanta, Atlanta. But now, two years later, I, I have.

00:04:10.000 --> 00:04:25.000

There's more data available. Also, I have a better understanding of the whole processing, and that's why I like to give another presentation today.

00:04:25.000 --> 00:04:44.000

Okay. The second part of his definition. I'm not going to read this, just like to point out the NLP is a soft field of linguistics, computer science and artificial intelligence, and it.

00:04:44.000 --> 00:04:55.000

Try to deal with the natural language data, natural language data means just the text, as if if he's English he just plain text.

00:04:55.000 --> 00:05:13.000

Also, the natural language data is unstructured the data. The compared to the structured data. So in the real world, most of their data is unstructured data about the machine learning.

00:05:13.000 --> 00:05:22.000

The main thing here is build a model based on the data. We also call the training data in order to make a prediction.

00:05:22.000 --> 00:05:25.000

That's a machine learning, purpose.

00:05:25.000 --> 00:05:34.000

So, you may ask, what is the relationship between an LP, and a ml.

00:05:34.000 --> 00:05:44.000

Basically, nowadays, but an LP existed long before the machine learning.

00:05:44.000 --> 00:06:04.000

Yeah. It is also called something like that text mining or text analytics. Look, buddy. ways the machine learning come along and LPs and right now is much more powerful and still NLP usually involves machine learning and AI and machine learning styles

00:06:04.000 --> 00:06:21.000

a half to involve NLP because machine learning can deal with our data structure data for example, that's the relationship between this tool.

00:06:21.000 --> 00:06:27.000

Some applications to offer an LP and the machine learning.

00:06:27.000 --> 00:06:32.000

I just gave some examples here.

00:06:32.000 --> 00:06:38.000

sentiment analysis, for example, social media.

00:06:38.000 --> 00:06:52.000

Pop modeling, or text summarization. This is a widely used in digital humanities. Nowadays, we say a lot of applications in this area.

00:06:52.000 --> 00:07:10.000

Another is a classification, and the categorization email spam filter is a typical example of this application. And my project belongs to this category.

00:07:10.000 --> 00:07:14.000

There is a speech recognition

00:07:14.000 --> 00:07:19.000

and a voice assistant, and the chat box.

00:07:19.000 --> 00:07:31.000

auto correct under auto competition. For example, prima or I do have a Primo here, listed search box. I don't need, we don't need to say more about this because we are so familiar with this part.

00:07:31.000 --> 00:07:41.000

So these are all the real world applications of NLP and, and others.

00:07:41.000 --> 00:07:49.000

As there are two types of machine learning, in general, supervisor supervised learning.

00:07:49.000 --> 00:07:53.000

For example, emails, and what does this mean is.

00:07:53.000 --> 00:08:13.000

Before we feed our data to the model, we label each data, we label them. For example, if he's an email spam, spam filter. Will he will label each email this is a span.

00:08:13.000 --> 00:08:35.000

model can learn from that. Learn from the labels, then the model can predict the new emails, that's supervised learning by the honor pipe lawful ml use unsupervised learning, for example they text the summarization because you faded a wish model without

00:08:35.000 --> 00:08:45.000

liberty, you don't you don't tell wishing water they are the machine will figure out machine the model will figure out at. Despite yourself, the key ideas.

00:08:45.000 --> 00:08:54.000

The summaries, often that the text.

00:08:54.000 --> 00:09:10.000

the pipeline. So the, what is the general steps, an LP and MLZ, they are basically those five steps from data gathering to pre processing.

00:09:10.000 --> 00:09:33.000

There are many manual steps involved in the pre processing and the vector ization on the model building, and the implementation. Yeah, in general, all the NLP and the machine learning, application involve this five general steps.

00:09:33.000 --> 00:09:55.000

So, basically, that's the foundations of the machine learning. Next slide. I'm going to go to the project itself, along the way I'm going to also explain some concepts, some ideas of this is special needs a pre processing and the vector ization.

00:09:55.000 --> 00:10:00.000

Yeah, I'll explain them along the way.

00:10:00.000 --> 00:10:06.000

So the first part is a gathering and the preparation.

00:10:06.000 --> 00:10:09.000

My project.

00:10:09.000 --> 00:10:23.000

I downloaded the initial questions from the chat transcript into Excel file. They are about close to 1000 questions.

00:10:23.000 --> 00:10:25.000

The next step.

00:10:25.000 --> 00:10:33.000

I have to apply and the garden economy IRB approval.

00:10:33.000 --> 00:10:49.000

Then I manually, I look at it this, the big spreadsheet manually to this thing. I removing blanks rows and also to remove the key repeated the questions experts or just clean it up.

00:10:49.000 --> 00:10:57.000

Yeah, bye bye just yet but my I ended my heart.

00:10:57.000 --> 00:11:11.000

Then the next big one is, I have a man you need label the each question as to this question is the reference question, or this class is no reference reference question.

00:11:11.000 --> 00:11:32.000

This take it may take the most of my time because imagine does more than 7000s of questions. You read each one of them and it will decide which which category, it is very labor intensive, and a time consuming.

00:11:32.000 --> 00:11:36.000

So this is a sample.

00:11:36.000 --> 00:11:41.000

After the manual clean up the data.

00:11:41.000 --> 00:11:50.000

The number is a question that under the label, I label them each one of them, and as a on the right is the question you dissolve.

00:11:50.000 --> 00:11:59.000

What I consider long reference question basically like a greeting only and always complain.

00:11:59.000 --> 00:12:08.000

We have a lot of noise can link questions, printing problems library service like IOL reserve I also consider them bound reference.

00:12:08.000 --> 00:12:15.000

Under also some of them are just not a question of for such as I'm sorry I was disconnected before. Yeah.

00:12:15.000 --> 00:12:19.000

So,

00:12:19.000 --> 00:12:29.000

I'm sure everyone may have their own opinion of what is reference what is nonprofits, but this is how I kick it.

00:12:29.000 --> 00:12:38.000

So, after the first step. The second step will use the one of the Worry, worry worried important step.

00:12:38.000 --> 00:12:40.000

Pre processing.

00:12:40.000 --> 00:12:45.000

This staff is essential in building a machine learning model.

00:12:45.000 --> 00:12:58.000

Basically, it transforms the raw text into a more digestible form so that and machine learning algorithm can perform better and achieve the result we want.

00:12:58.000 --> 00:13:07.000

So basically it evolved for first the five amassed the six is optional.

00:13:07.000 --> 00:13:25.000

The first one first, the two, I don't want it no needed to say more. You're not just worried that understand what is tokenization tokenization basically is a simulated their sentences into a list of individual words, basically means like, remove in the

00:13:25.000 --> 00:13:29.000

spaces between words. That's a tokenization.

00:13:29.000 --> 00:13:42.000

Also, we needed to remove stop words like is the. And we, the purpose of this is, you want to only keep as a pivotal, the meaningful words for the model.

00:13:42.000 --> 00:13:57.000

So the model can perform better to predict the to predict to distinguish between reference question, or non reference question we want to go to all those pivotal and the meaning meaningful words.

00:13:57.000 --> 00:14:02.000

later on our show your stop list.

00:14:02.000 --> 00:14:10.000

There's a stop list of more than 100 of them. That's what I use the My, my.

00:14:10.000 --> 00:14:26.000

Problem number five is called a spamming or memorizing this is important apart because we want a chance to the semantic meaning of move off related words.

00:14:26.000 --> 00:14:31.000

In other words, you explicitly correlates words with a similar meaning.

00:14:31.000 --> 00:14:43.000

For example, run, running, and runner. After this standing, it becomes Justin One, two we group them together to find a

00:14:43.000 --> 00:14:46.000

single word.

00:14:46.000 --> 00:15:02.000

So the model, not only to, to save a model, and save this time and effort for the model, because I didn't reduce a lot of work. Also, because for the marble bushy model.

00:15:02.000 --> 00:15:15.000

It doesn't know the difference between you and the running for us as human. We think we know it's related to work but for the, for the machine, it is just consider running the running are totally two different things.

00:15:15.000 --> 00:15:27.000

So, after this standing. It will help them to for the, for their performance library under libraries become library.

00:15:27.000 --> 00:15:35.000

And is this stemming is kind of like a boot for chop off the ending of of each word.

00:15:35.000 --> 00:15:49.000

And the next one is memorizing because of gurus and the keys, if we use the spamming, it cannot to do it, it just to get rid of, he left the ways of Google energy.

00:15:49.000 --> 00:16:01.000

That doesn't make any sense butterfly memorizing it will find the sentiments basically that they're limiting no gurus and the key Islam refers to the same root word.

00:16:01.000 --> 00:16:23.000

Basically, they monetizing always return on dictionary word dictionary word, but steaming is a brutal they just chop off the nd or suffix. So, there's trade off also stemming is fast, but the limit sizing is more accurate, are these more resources, using

00:16:23.000 --> 00:16:28.000

more resources.

00:16:28.000 --> 00:16:41.000

The last point six this staff who is a feature engineering feature engineering is optional.

00:16:41.000 --> 00:16:46.000

Basically, is try to create a new feature. What is it a feature.

00:16:46.000 --> 00:17:09.000

The feature basically the member that the spreadsheet. If we have a spreadsheet, and we have rows and columns, the role the roles, basically, is the each of the questions that is the columns columns that uses a each individual word is a feature.

00:17:09.000 --> 00:17:22.000

So we want to create a new feature. And I just assumed like a year for the length of the reference question or the long reference question, or somehow defend.

00:17:22.000 --> 00:17:35.000

So I can create a new feature of question lens, and hope it will help our model distinguish between references to help them to distinguish the reference question or the long reference question.

00:17:35.000 --> 00:17:40.000

Now today will feature engineering. The purpose of that.

00:17:40.000 --> 00:17:48.000

So I use this histogram, to find out if there's actually difference between the two.

00:17:48.000 --> 00:17:52.000

Two types of questions, ladies.

00:17:52.000 --> 00:18:00.000

You can see the blue collar distribution and use it for reference question.

00:18:00.000 --> 00:18:15.000

And as a yellow color is a for long reference question, it doesn't show the difference here between these two groups. The trend, each of the reference question is somehow the, the characters is more is longer.

00:18:15.000 --> 00:18:28.000

There's more character in a reference question and non reference question is shorter. There's some somehow we can tell.

00:18:28.000 --> 00:18:31.000

Okay, now come to the Python code.

00:18:31.000 --> 00:18:45.000

And I use the Jupiter notebook. And we imported this packages, I'm just going to go Preakness through this code to explain this is the first part of my problem.

00:18:45.000 --> 00:19:11.000

I used a stammer is called a porter stammer for for my stemming. So, usually you just use one or another you don't use both. If you use a standard standing, you are not using them.

00:19:11.000 --> 00:19:17.000

a new feature, right here, the lines question is not a new feature I created.

00:19:17.000 --> 00:19:28.000

Then next part is the pre, the pre processing partner. This is three lines of code here deed.

00:19:28.000 --> 00:19:34.000

Remove punctuation here. The first one, tokenizing, right here.

00:19:34.000 --> 00:19:47.000

Then lastly, to remove the stop words. This week three lines of code.

00:19:47.000 --> 00:19:58.000

Okay, so this is a result of the pre processing and the feature engineering from the left to right.

00:19:58.000 --> 00:20:04.000

This is question number here from all the way from the one to 7000.

00:20:04.000 --> 00:20:15.000

Plus, and is a label. Yeah, I labeled them each one of them up to reference on your reference. And this is the original question.

00:20:15.000 --> 00:20:25.000

And this is the new feature I created for each each question there's the length of the character for each question.

00:20:25.000 --> 00:20:34.000

Somehow you can see reference question to be could be, yeah, that's a longer. Yeah, to be longer yeah here.

00:20:34.000 --> 00:20:52.000

And after the Remove punctuation. This is the become this part. After lowercase, and the tokenizing.

00:20:52.000 --> 00:20:57.000

So, for example,

00:20:57.000 --> 00:21:00.000

books become book. Yeah.

00:21:00.000 --> 00:21:08.000

You can see the, the, the here. Yeah.

00:21:08.000 --> 00:21:15.000

Let's stop word so list, I used their total of 179.

00:21:15.000 --> 00:21:22.000

Yeah, often

00:21:22.000 --> 00:21:27.000

Holly's my time do it.

00:21:27.000 --> 00:21:28.000

Okay.

00:21:28.000 --> 00:21:41.000

Um, what is the vector ization. The purpose is to transform the text data into numerical data so that the machine algorithm and kind of pies and understand the use that data to build a model.

00:21:41.000 --> 00:21:53.000

So remember that. Can you not always deal with numbers. Yeah, they don't understand what is the text means we have to fight for more the text into the numbers.

00:21:53.000 --> 00:21:57.000

Yeah. So, Hollywood is conducted.

00:21:57.000 --> 00:22:27.000

We have to build a vector of new numerical features, not to present the sun object. For example, we have, if we have two questions here. Hello. What are the library farmer, our the sake of the ones that can I print the color in the library and if so how

00:22:29.000 --> 00:22:31.000

And

00:22:31.000 --> 00:22:46.000

so, they it a unique words, the eight unique words right here from Hello lot library farmer, our friend the color, our unique words, so we can't.

00:22:46.000 --> 00:23:01.000

Each word from each question, the frequency of how many times it obscured the inside, in this, in this whole, the table right here, we just counted them simply counted them.

00:23:01.000 --> 00:23:08.000

Yeah, if we appear once with the one if he's up here zero, we put zero.

00:23:08.000 --> 00:23:10.000

That's how does.

00:23:10.000 --> 00:23:27.000

Basically, there are many types of lateralization the butter the most popular three count vector riser and grants and TF IDF.

00:23:27.000 --> 00:23:37.000

This is a extremely simplify the example of what is a machine learning, working behind.

00:23:37.000 --> 00:23:47.000

By doing this, For example, the ID is our question you for we just took a two words, one is article wines the print.

00:23:47.000 --> 00:24:06.000

And for example if you question number one, they were article beat up here twice, and the printer, they were up here zero times, and so on so forth, and is the label will say this quote number one is a reference question.

00:24:06.000 --> 00:24:24.000

And number four is non reference question. So, from this we can see, if we feed this data to the model. The model will quickly realize that, or if the article is closely related to reference question.

00:24:24.000 --> 00:24:32.000

And in the word the printer is closely related to non reference questions. That's how the machine learned.

00:24:32.000 --> 00:24:54.000

But remember there's thousands of this Collins, so the relationship is quite complicated. This is just for illustration purposes, and also this is a simply count vector riser, where is the most simple way of doing this, this, this is very synchronous

00:24:54.000 --> 00:24:58.000

for.

00:24:58.000 --> 00:25:00.000

Okay.

00:25:00.000 --> 00:25:11.000

I use the TF IDF wax riser because it's just, it's a weighted, and is more accurate than count vector riser.

00:25:11.000 --> 00:25:24.000

The under the resolve, is we have more than 7000 rows, that is a questions, all the way down, and the unique words is 6872 columns, right here.

00:25:24.000 --> 00:25:32.000

And you can see inside, most of them just zeros, but some of them do have values here.

00:25:32.000 --> 00:25:36.000

Right here inside. So this is a vector rise.

00:25:36.000 --> 00:25:54.000

Lots of the data we faded to the model to the model is performed on this. That's a whole, everything is working behind the behind the behind the scene.

00:25:54.000 --> 00:26:00.000

Okay, Now come to the model building either evaluation.

00:26:00.000 --> 00:26:07.000

I use the two popular models, I just want to compare the result and see which one's better.

00:26:07.000 --> 00:26:24.000

The most to popular wise is random forest, and the gradient boosting model, the random forest the model is the basic needs of building, many many decisions which, at the same time is a parallel computing, and the desired by majority will.

00:26:24.000 --> 00:26:33.000

And the gradient boosting is a building one decision to a one at a time. So each new to help to protect the arrows made the back radius and it's mandatory.

00:26:33.000 --> 00:26:40.000

And the boosting by reward and the penalty.

00:26:40.000 --> 00:26:56.000

Okay, before we go into actually building the model, I want to introduce this confused confusing matrix, this is basically we use this to evaluate the performance of our model.

00:26:56.000 --> 00:26:59.000

Yeah.

00:26:59.000 --> 00:27:22.000

Basically, there are three parameters to to evaluate is a performance, accuracy, which is to positive would Plaza to negative divided by this or I did have together accuracy and precision.

00:27:22.000 --> 00:27:41.000

This is a tool positive with divided by two polity will plus the false positive and the recall the three parameters. Yeah. So, the key is, want to use consider to be positive, I know what would be negative.

00:27:41.000 --> 00:28:00.000

I know this is a very confusing and but next slide, I'm going to use using on my project example to explain, and the, if you will. Yeah, either to understand, after that.

00:28:00.000 --> 00:28:09.000

In my case, I consider the yes label the reference reference question to be positive. So the no label is negative.

00:28:09.000 --> 00:28:20.000

So, the accuracy is become correctly predict the question, divided by total this precision is this.

00:28:20.000 --> 00:28:23.000

Yeah, and a recall.

00:28:23.000 --> 00:28:30.000

Okay, that's still confusing but now, if we have a total 100 questions.

00:28:30.000 --> 00:28:43.000

And we predicted 90 of them correct me. Remember we predict 90 corrected me, which include the line he cracked and he will include the 60s reference question, and the third is non relevant questions.

00:28:43.000 --> 00:28:47.000

But before the time questions we predicted one.

00:28:47.000 --> 00:29:04.000

There are two RAF question, we predict we predict is non graph, and eight longer if we predict the RAF. Okay, so here we go. So what is the accuracy. The accuracy is 90, divided by 100 x 0.9.

00:29:04.000 --> 00:29:11.000

The precision precision is 60 divided by 60 plus two.

00:29:11.000 --> 00:29:16.000

So, which is 0.96, and the recall. Here is recall.

00:29:16.000 --> 00:29:21.000

So, which one is more important.

00:29:21.000 --> 00:29:37.000

Precision or recall that depend on the real problem in our case of precision is more important. We want to as large as possible. Why, because when we predict.

00:29:37.000 --> 00:29:49.000

Non reference question as reference question which is not a serious, but if we probably pretty the model predict the reference question, as a norm referenced question.

00:29:49.000 --> 00:29:52.000

I think the consequences more serious.

00:29:52.000 --> 00:30:06.000

So we want to decrease the amount of incorrect and predict the long reference question that's, that's why we think of proceed precision is more important than in our case.

00:30:06.000 --> 00:30:12.000

I hope that you explained better clearly very confusing.

00:30:12.000 --> 00:30:20.000

Okay, now we start to build our random forest model.

00:30:20.000 --> 00:30:23.000

This is a Java and Python code.

00:30:23.000 --> 00:30:32.000

And just pay attention to. Only I want to point out is the last added the Barton, the result precision.

00:30:32.000 --> 00:30:44.000

0.91 for recall and accuracy. This these ley line here we got from this model.

00:30:44.000 --> 00:31:14.000

And for green boosting model building the precision is 0.409904, the recall, and accuracy, I would say, all those three numbers, the random forest model, perform perform better than, than the gradient boosting model.

00:31:14.000 --> 00:31:25.000

So they resolved. I would choose the random forests the classifier the model, because these, you have better values, perform better.

00:31:25.000 --> 00:31:35.000

But there are two important concepts or ideas I escaped.

00:31:35.000 --> 00:31:39.000

They are considered, otherwise the technique.

00:31:39.000 --> 00:32:00.000

For the sake of time I didn't go into detail why is the cross validation, or nada is the greatest search. What is the crossover divided nation. The cross validation basically is divided our Dave has said, Remember we have close to 8000 questions.

00:32:00.000 --> 00:32:10.000

And we divided into basically like four sets, each side to have 2002 questions, and the Philosopher's for facts.

00:32:10.000 --> 00:32:32.000

We cycle them each time we use one set of data as a testing set. The other three sets of questions data, we use the for the tweet training set. So which we, we fade it also three sides to the model.

00:32:32.000 --> 00:32:37.000

Then we test the use one set and the tester our prediction.

00:32:37.000 --> 00:32:42.000

So you just the cycle then each each set us.

00:32:42.000 --> 00:32:48.000

Eyes Alliance as testing, and the three times as training.

00:32:48.000 --> 00:32:53.000

So you cycle them that lottery is a cross validation, or technique.

00:32:53.000 --> 00:33:03.000

So what is agreed search query the search is basically for each model.

00:33:03.000 --> 00:33:08.000

There are just lots lots of all super parameters.

00:33:08.000 --> 00:33:34.000

For example, the an estimator here where you can you can you can define them is basically like how many tweets you want to build. You can build a like your country's 1520 5100 150 staff, those are called the super parameter, they are just lots of them

00:33:34.000 --> 00:33:50.000

for one model I know we can adjust the YouTube parameter to save all the to perform. So the combination of all the parameters, and also the cross validation.

00:33:50.000 --> 00:34:09.000

This leads to build up agreed with running each each sale to come together to resolve under the, the, the model were generated, or resolve for each sale of this great this great is a combination of each cycle of the cross meditation practice our super

00:34:09.000 --> 00:34:17.000

parameters, we have arranged for each filter parameter. We said we said in our code, basically.

00:34:17.000 --> 00:34:33.000

So ladies and model run itself and come out with the optimal resolve, because we don't want to overfeed, also we want to oppressive enough, but not over aggressive.

00:34:33.000 --> 00:34:36.000

This is a yes I'm seeing the.

00:34:36.000 --> 00:34:49.000

The. Sometimes he's getting very very tricky. That's also you need to spend a lot of time on decide on this, and just escaping this part This, I think, is to have the last part.

00:34:49.000 --> 00:34:52.000

Yeah.

00:34:52.000 --> 00:35:19.000

Okay, so the next step. Remember the five steps, implementation, implementation, implementation, I didn't do it in my library is.

00:35:19.000 --> 00:35:45.000

many chats. And also, it needed to be integrated into some chat application for example like a deeper answer deep chat integrated or better maybe return eyes are plotting or add on to some kind of chat problem us as a filter.

00:35:45.000 --> 00:35:59.000

Yeah, I think that's the possible future implementation for this for this program. Maybe I'll, I'll write a write one you feel it for myself. Yeah.

00:35:59.000 --> 00:36:16.000

But does the next thing I can do, because this is a simple adjust that differentiate between reference and the long reference. But the next icon what we can do is we can build a multi class model to classify the questions into multiple categories, for

00:36:16.000 --> 00:36:34.000

example, we can have an article search category Book Search noise complain library services, and the real span, etc, like five to six categories. So the model will predict or distinguish between them.

00:36:34.000 --> 00:36:44.000

That's what my, my idea, I think, you know, next scene can do.

00:36:44.000 --> 00:36:51.000

So I want to do a quick recap with with with, we talk about the fundamentals.

00:36:51.000 --> 00:37:05.000

the types of machine learning, application of the sentiment sentiment analysis classification topic the modern era virtual also the pipeline, the five steps.

00:37:05.000 --> 00:37:27.000

The data carry pre processing. Remove punctuation, change it to lowercase, remove stop words tokenization stemming or memorizing an official engineering optional, and a vector ization, we talk about the current I gave a very simplified illustration of

00:37:27.000 --> 00:37:30.000

the how they count.

00:37:30.000 --> 00:37:47.000

Vector ization works for the motion model and the way I actually use TF IDF counter, a vector riser, and the the final needs a model building, and the project who I used it, I tried the tool models and decided on us they choose random forest model is

00:37:47.000 --> 00:37:58.000

I tried the tool models and decided on us they choose the random forest model is better than the gradient boosting.

00:37:58.000 --> 00:38:08.000

And then also, I talked about the evaluation matrix, the accuracy, precision and recall.

00:38:08.000 --> 00:38:13.000

I think that's all I had.

00:38:13.000 --> 00:38:17.000

But I have a question for everyone to think about.

00:38:17.000 --> 00:38:27.000

So well someday the chatbots supplement, or even replace the real people to provide library virtual service in the library.

00:38:27.000 --> 00:38:38.000

I don't know the answer, you think if we train, we have lots of data available. If we train as a machine learning model.

00:38:38.000 --> 00:38:48.000

I think it is possible. Yeah, to either nice to to do some partially to to do something, actually.

00:38:48.000 --> 00:39:08.000

I don't know if in the library field but in order for the industry fail. I think the robot is already there, to, to answer people's question, but maybe some, some library or some lenders are already thinking about is developing Samsonite, you know, I

00:39:08.000 --> 00:39:15.000

I don't know, but the just question left for people for us to think about this.

00:39:15.000 --> 00:39:20.000

Okay, that's it. Thank you very much.

00:39:20.000 --> 00:39:31.000

Thank you very interesting presentation, we have not been following the chat but there's been a number of comments that were impressed about your analysis and the process that you undertook.

00:39:31.000 --> 00:39:35.000

And there was one question that came in, just toward the end.

00:39:35.000 --> 00:39:40.000

In the label data set how many samples were no income anywhere. Yes.

00:39:40.000 --> 00:40:01.000

Oh, I didn't have it you can stop number but the I think yes he's more about auto more than 7000. I think there's about as close to five solid is about 2000 or no.

00:40:01.000 --> 00:40:06.000

I do have a question, if another library was to try to undertake a similar project.

00:40:06.000 --> 00:40:20.000

Would they be able to take advantage of any work that you've done or other libraries have done to save some of that upfront work, but yeah so Yes, certainly.

00:40:20.000 --> 00:40:34.000

here. No problem. Just in general, if you've gone through that process of identifying the words that are associated with reference versus non reference it seemed that that would be applicable to other you know sets of data as well.

00:40:34.000 --> 00:40:50.000

Yes, certainly. But it's better to use other libraries own data, I think, because each library do have their unique, I think the difference between libraries.

00:40:50.000 --> 00:41:04.000

But they have common I think the most of most of them are same, same either the content, the questions, others they do have different, I think your stuff.

00:41:04.000 --> 00:41:14.000

Just one comment came in, maybe we could help filter questions and answer the easy ones and mark the ones needing follow up by a person that was in response to your question at the end.

00:41:14.000 --> 00:41:34.000

Yes, I saw that I think that's good idea. Actually I'm thinking, if we used to like, if we use like a printing or il question maybe we have a department IoT parliament and also have this chat and then they filter will automatically directed that question

00:41:34.000 --> 00:41:43.000

to the IoT parliament. So, without even the reference librarian said, that's, Yeah, right.

00:41:43.000 --> 00:41:50.000

Yes, all day yeah automatically answer to prevent SQUFAQ. That's correct, yes.

00:41:50.000 --> 00:41:59.000

So you become like a half automatic or maybe totally automatic and for some kind of question. Yeah, automatically answered.

00:41:59.000 --> 00:42:09.000

Yes.

00:42:09.000 --> 00:42:23.000

And these are questions that we have before we we wrap up for the day.

00:42:23.000 --> 00:42:33.000

Alright then I guess we'll say thank you again for our nation will be back tomorrow for the final day of Enoch.

00:42:33.000 --> 00:42:59.000

Okay. Thank you. Okay.