Video to Skill Tagging using Transcripts under Weak Supervision

Zhe Cui
Department of Electrical and Computer Engineering
University of Maryland, College Park
College Park, MD 20742
zcui@umd.edu

Shivani Rao
LinkedIn, Inc.
Sunnyvale, CA 94085
sirao@linkedin.com

Abstract

Delivering micro content (small pieces of learning materials) from a course catalog has become important as it helps learners engage quickly with a learning platform on a need-to-know period. In order to recommend videos that can teach skills relevant to a learner, we propose an approach for tagging videos with skills using transcripts. Since, there is no labeled data to train a video-to-skill model directly, we leverage a bunch of techniques that fall under the umbrella of weak supervision. We evaluate our approach using offline metrics like Precision, Recall, F-1 score and online metrics via large-scale A/B testing.

1 Introduction

Online Learning has taken off with a plethora of e-learning products such as Coursera, LinkedIn Learning, and Udacity, to name a few. More recently, learners are engaging with online learning platforms in an informal way, i.e., they are seeking relevant videos on a need-to-know period as opposed to taking on the task of sitting through an entire course. In order to deliver relevant videos to the learner (whose learning interests are expressed in terms of skills), we need to map videos to skills, so that we can match learners and videos in the skill space. The LinkedIn Learning platform [2] that we focus on in this paper hosts thousands of courses and each course comprises of several videos, resulting in hundreds of thousands of videos, and it is impossible and expensive to have Subject Matter Experts (SMEs) hand-label all the videos. Even at the course-level, the SME provided skill-tags are sparse and incomplete and cover only 2% of the skills on the platform.

In this paper, we propose an approach that utilizes raw transcripts to tag videos with skills for the purpose of recommending micro contents to learners on LinkedIn Learning platform. One of the byproducts of our proposed algorithm is that, it also improves the quality of course-to-skill mapping as well, i.e., the skill coverage grows by 3X for the mapping. We present our experimental findings using offline (e.g., Precision/Recall) and online metrics (e.g., A/B testing).

2 Related Work

In industry settings, getting enough labeled training data has always been a bottle neck, but academic research focus on this problem has been more recent [19]. This form of learning through videos is called micro learning, and the videos are called micro content in educational research community [15, 16]. Some of the techniques that have been developed to circumvent this problem include active learning [18, 7], semi-supervised learning [5, 17, 10], and transfer learning [14, 4] [1]. In the field of deep learning, unsupervised pre-training [8] has been used as proven technique to improve

[1] Due to page limit, please refer to the cited paper for a full survey of related work.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

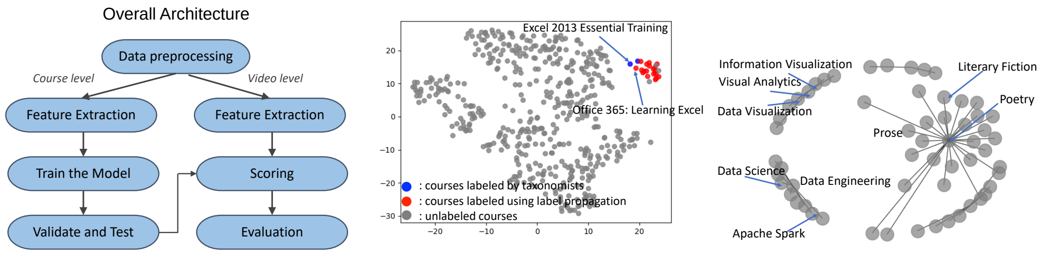

Figure 1: Left: proposed model architecture. Middle: Visualization of course representations (with PCA and t-SNE [13]). We propagate skill labels to courses that are close neighbors of a labeled course in this embedded space. The skill label is propagated from two courses related to excel (blue) to other courses (e.g., Office 365.). Right: subset of skill graph. There are three clusters shown: Data Engineering, Poetry, and Commerical Design. Each of them are closely related to similar skills e.g., Data Science to Data Engineering, Literary Fiction to Poetry.

performance of classification tasks especially in the context of transfer learning and semi-supervised learning. Our work is inspired by all of these contributions, and stands to show how they can be applied to solve a data problem in the industrial setting.

3 Modeling

The problem we are trying to solve can be described as follows, "Given video transcripts, infer skills on the learning platform". Since the set of skills is derived from a standardized dictionary [3], our approach of extracting skills is to build a separate text classification model for each skill. Thus for each skill, the feature matrix remains the same across models, and the labeled column changes. While extracting tags or assigning skills to textual information is not new, there are some unique challenges for skill tagging of video transcripts in the context of learning from limited labeled data.

- *Videos do not have skill labels.* The task of labeling videos with skills is an unsupervised learning task, given no labeled training data available. Thus we create a model that can be trained on course level, by combining all the transcripts of the videos in that course into one document (Fig. 1 Left). This turns it into a transfer learning problem [14]. Since the model is trained entirely on textual features, we can use the same model to predict skills for video transcripts.
- *For each course, the number of skills tagged by SMEs is sparse (2-3 per course).* In order to alleviate the issue, we use a heuristic version of the label propagation technique to expand the labels [20].
- *Existing course level skill labels only cover 2% of the total skills.* The purpose of our proposed approach is not just learn a robust video to skill mapping, but also discover *new* skills that are relevant to a course or a video, that was probably missed about by SMEs. In order to discover new skills, we expand labeled skills using an external source that we call the skill graph [2]. based on correlated skill graph to generate more labeled skills for training.
- *Text features are very sparse.* We use unsupervised pre-training on the courses, by learning a doc2vec [11] representation of transcripts to create document embeddings.

The overall video to skill model flow is shown in Fig. 1 Left. The main idea is gained from transfer learning: **train at the course-level, and make predictions at video level**. Specifically, each course level transcript is obtained by concatenating video level transcripts and represented using latent representations (doc2vec [12], details below). We present key procedures of the proposed model below.

3.1 Label Augmentation

As mentioned before, the text classification problem is faced with a challenge of limited labeled data. We use two approaches to increase the labeled data for each of the per-skill models.

- **Course based expansion:** The expansion of skill labels for courses is based on course similarities. An example is shown in Fig. 1 Middle. Using the document embeddings of courses, each course pair can be measured by a similarity metric (here we use kNN kernel as the metric). The skill label “Excel" has been expanded to other courses that are close in the graph. This step can be considered as a heuristic version of label propagation [20].
- **Skill based expansion:** We leverage an external source of skill graph [1] to increase the labeled training data. A subgraph is shown in Fig. 1 Right. The weight of the edges represents similarity [6]. In order to ensure the quality of the augmented labels, for each skill, we retain only the skills that have similarity score in the 99th percentile (top 1%). This step can be considered as mixture of noisy labeled data with high precision labeled data from the SMEs.

3.2 Model Training

With all the labeled data (including those obtained with label expansion), we build a “per-skill” model: training the model separately for each associated skill. In spite of applying all the label generation schemes discussed in the last section, the percentage of labeled data is still small. Since the model needs to score and rank skills, it is preferred to get predictions with quantitative measures to rank them. Hence, we chose logistic regression [9] which gives a score for each skill prediction between 0 and 1.

3.3 Scoring Pipeline

Once we have trained per-skill models, we can score any video or course to predict which skills are relevant. In what follows, we present two key considerations while designing our scoring pipeline:

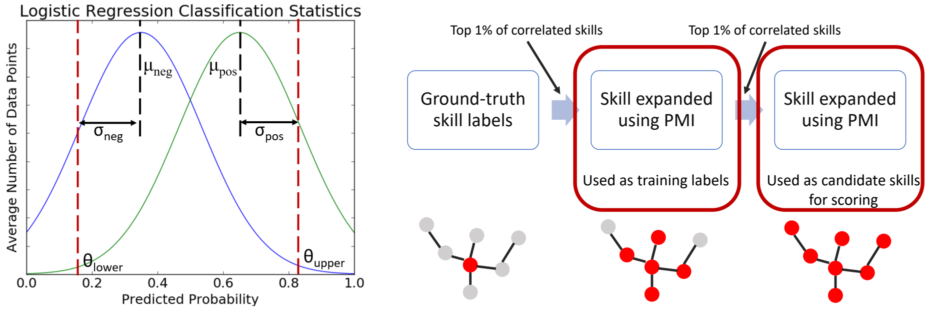

Figure 2: Left: one sample model with predicted distribution of positive and negative labels. Truncate at θ_{upper} to retain only confident skills. Right: skill expansion based on skill-graph, for the purpose of generating candidates. During training, we only did 1-hop expansion (See Figure 1 Right) to generate labels. For the purpose of scoring, we use 2-hops to generate candidates for scoring.

- *Preserve confident predictions.* The goal is to retain only skill predictions that the model is confident in. Typically, logistic regression uses 0.5 as default threshold, but for our model, we set the threshold to θ_{upper} and θ_{lower}, which can be set heuristically as $\theta_{upper} = \mu_{pos} + 2 * \sigma_{pos}$ and $\theta_{lower} = \mu_{neg} - 2 * \sigma_{neg}$ respectively.(Fig. 2 Left[2]). With $probability$ outside the range $(\theta_{lower}, \theta_{upper})$, the predictions are considered to be confident.
- *Candidate Pool Generation.* There are thousands of different skills, which means thousands of models trained for predictions. If we were to predict every course and video against all the skills, it would be rather inefficient (e.g: there is no point in scoring a piano course transcript for the skill Java) and we would also run into having lots of false positive predictions. Hence, for each of the courses/videos we only evaluate skills that are possibly relevant or related to a given skill that is already tagged by SMEs. We utilize the skill graph for this, but this time, we use a 2-hops to retrieve our candidates. Fig. 2 Right gives an illustration of such a two-step procedure using the skill graph.

[2] To keep confidentiality, we remove the actual numbers on Y-axis of the left figure.

4 Evaluation

While the primary purpose of the proposed model is to map videos to skills, which helps video recommendations, to test the efficiency of the approach, we evaluate both course to skill and video to skill mappings. For video to skill mapping, we are collecting feedback from our SMEs as we speak (For some examples of video-transcript to skill mapping, please see Appendix). However, we do have mechanisms to do large-scale online A/B testing with our course-to-skill models as we will describe in this section.

4.1 Datasets and Metrics

We apply our proposed model to over $5k$ courses and over $100k$ videos. Typically, one course is labeled manually with 1 or 2 skills by SMEs. With skill expansion using the skill graph, the number of unique skills is tripled. The metrics we use include: Precision@K, Recall@K, F-1 score@K, as well as Click-Through-Rate (CTR), which is online metric that measures the number of engaged learners.

4.2 Research Questions

We present our evaluations according to the following research questions:

How does changing the threshold θ_{upper} in logistic regression model affect the offline metrics?

Fig. 3 Left shows a skill model of Precision, Recall and F-1 score changes across different probability thresholds. The Recall stays high most of the time, while Precision keeps increasing as threshold goes higher. From a practical perspective, a course cannot teach too many skills, so it is reasonable to err on the side of caution and focus on Precision, not Recall.

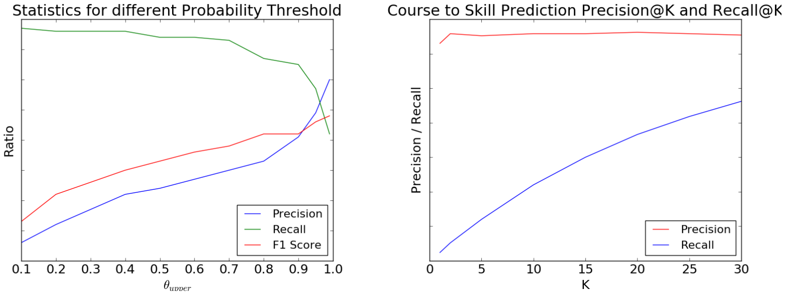

Figure 3: Left: Precision and Recall with different probability thresholds on predictions from one course to skill model. Right: Precision@K and Recall@K for top K retained skills of the proposed approach.

How does the performance change w.r.t. K, the number of top skills (K) we retain?

Fig. 3 Right gives an overview of how Precision and Recall changes w.r.t. the number of top predicted skills we retain. It's worth noting that while Recall is an important metric in most classification and retrieval scenarios, it can be artificially low for course to skill mapping where SME provided labels are sparse and incomplete. This is due to the fact that SMEs are unable to retrieve all the relevant skills for a given course. Typically each course has 2 or 3 labeled skills from SME, and the number of predicted skills for each course may increase up to 40 or more. For example, "Python" course may get assigned more than 80 skills since it is a fundamental course for several types of skills such as "Web development", "back-end development", "Data Science", "Machine Learning"..

Is it possible that the same skill tags will be assigned to all the videos in a course?

Since we train the model based on course level skill labels, it's possible that all the videos within the same course get the same labels. Thus measuring the skill variance for videos within a course gives us valuable information about how the model is able to differentiate between video level features. Specifically, skills that are predicted for majority of the videos in a course can be bubbled up to a course level skill mapping, and the skills that assigned to only some videos will be kept as actual video labels.

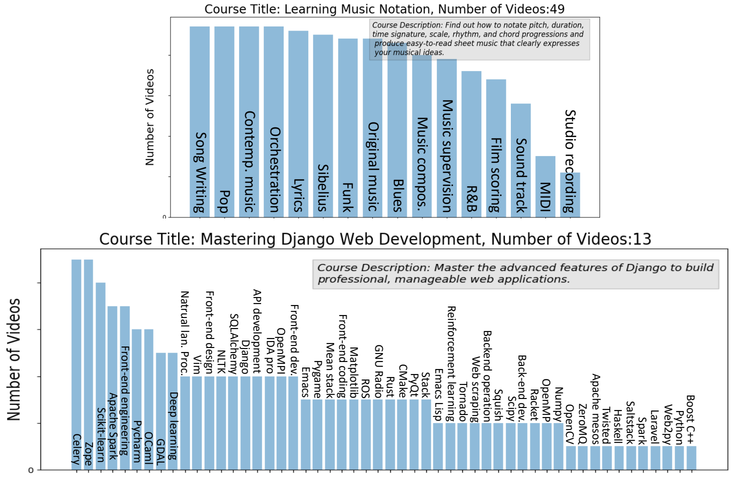

Figure 4: The number of predicted skills appears in each video within the same course. Course on the top has a small variance while variance of predicted skills of the bottom one is larger.

Fig. 4 shows video to skill variances for two typical courses. It shows the number of videos tagged with each skill. The “Learning Music Notation" course has very low skill-variance, whereas the “Django Web Development" course has a higher variance across videos.

How does the predicted course-to-skill mapping compare to the existing course-to-skill mapping in Online A/B testing?

In order to measure the performance of our predictions, we have run a large-scale online A/B testing experiment on one of our engagement channels – the jobs page. Job seekers arrive on this page to view jobs and possibly apply for jobs. For existing learners, this page serves as another channel where they can be directed to the learning platform by showing them course recommendations relevant to the job they are viewing. The jobs themselves map to skills, and with courses mapping to skills as well, jobs can be matched with courses in the skills space. By keeping the job-to-skill model the same, we created two job-to-course models using two course-to-skill mappings: (a) j001 - job-to-skill (b001) matched with course-to-skill model (where skills are tagged by SMEs) and (b) j002 - job-to-skill (b001) matched with course-to-skill model (where skills are tagged by our approach). We doubled the job-to-course coverage from j002 to j001 (meaning 2x more jobs had course recommendations due to the higher skill coverage of the proposed course-to-skill model). On the jobs page, we saw 65% increase in impressions, while keeping the Click Through Rate (CTR) unchanged (which means, that we increased coverage (read-Recall) without hurting Precision). Last but not the least, with j002, there is 4.6% increase in learner engagement (unique learners that watched course videos) on the learning platform page.

5 Conclusion

In this paper we propose an approach to extract skills from video transcripts with no labeled data. We drew inspiration from transfer learning, in that, we trained a course-to-skill model by using textual features only and apply this model to predict skills for the videos. While we started with the goal of extracting skills from video transcripts, our approach also improves course to skill mapping. While we are still collecting feedback on the quality of our video-to-skill recommendations from SMEs, we used offline metrics and online A/B testing experiments to gauge the quality of course-to-skill predictions made by our model. For future work, we plan to investigate with more sophisticated approaches to learning document representations from text. We also plan to use different sources of data for our skill-to-skill mapping. We plan to use the prediction score to intelligently derive a course-skill and video-skill pairs for which we need to collect feedback from SMEs (tending towards an active learning approach). While, we acknowledge that this paper itself does not propose many novel ideas, we argue that the work done here shows the power of weak supervision approaches by applying them to an industry problem, where labeled data is indeed hard to come by.

References

[1] 2017. LinkedIn Economic Graph Research. (2017). https://engineering.linkedin.com/data/economic-graph-research.

[2] 2017. LinkedIn Learning. (2017). https://www.linkedin.com/learning/.

[3] 2017. LinkedIn Skills List. (2017). https://www.linkedin.com/directory/topics-a/.

[4] Isabelle Augenstein, Andreas Vlachos, and Diana Maynard. 2015. Extracting relations between non-standard entities using distant supervision and imitation learning. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 747–757.

[5] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. 2009. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks* 20, 3 (2009), 542–542.

[6] Thomas M Cover and Joy A Thomas. 2012. *Elements of information theory*. John Wiley & Sons.

[7] Gregory Druck, Burr Settles, and Andrew McCallum. 2009. Active learning by labeling features. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*. Association for Computational Linguistics, 81–90.

[8] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. 2010. Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research* 11, Feb (2010), 625–660.

[9] David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. 2013. *Applied logistic regression*. Vol. 398. John Wiley & Sons.

[10] Samuli Laine and Timo Aila. 2016. Temporal Ensembling for Semi-Supervised Learning. *arXiv preprint arXiv:1610.02242* (2016).

[11] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*. 1188–1196.

[12] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*. 1188–1196.

[13] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* 9, Nov (2008), 2579–2605.

[14] Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22, 10 (2010), 1345–1359.

[15] Cristobal Romero and Sebastian Ventura. 2007. Educational data mining: A survey from 1995 to 2005. *Expert systems with applications* 33, 1 (2007), 135–146.

[16] Cristóbal Romero and Sebastián Ventura. 2010. Educational data mining: a review of the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 40, 6 (2010), 601–618.

[17] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*. 2234–2242.

[18] Burr Settles. 2010. Active learning literature survey. *University of Wisconsin, Madison* 52, 55-66 (2010), 11.

[19] Weak Supervision: The New Programming Paradigm for Machine Learning 2016. Weak Supervision: The New Programming Paradigm for Machine Learning. (2016). https://hazyresearch.github.io/snorkel/blog/ws_blog_post.html.

[20] Xiaojin Zhu and Zoubin Ghahramani. 2002. Learning from labeled and unlabeled data with label propagation. (2002).

Appendix

Here we provide two examples of video transcripts, corresponding course title and descriptions, as well as predicted skills using the proposed model.

5.1 Example 1: A Java Course

This example shows video transcripts from a Java course and corresponding predicted skills. The order of the skills is based on the predicted probabilities.

Course Title: Learning Java Applications (2012)

Course Description: An introduction to developing Java applications for various runtime environments.

Video Transcripts: Now we'll write the code to connect to our database, so make sure first that your database server is running and that Tomcat is running inside of Eclipse and that you have installed the appropriate MySQL driver. So I'm in data.jsp. I am going to double-click the tab so that I can get a full view of just my code. You can always double-click it again to go back to the standard view. The first thing we are going to do is write some import statements. So open bracket, a percent sign and that symbol, we will close that out with a percent sign and a greater than sign. And inside of this tag I'm going to type page, space, import, an equal, and in quotes I am going to type in the classes that I want to import, so java.sql.*. And then I'm going to copy and paste this code right after itself to eliminate whitespace. So I'll paste that using the keyboard shortcut Command+C or Ctrl+C to copy and Command+V or Ctrl+V to paste. And I'm going to change the import code from java.sql.* to java.io.*. And I will repeat the process and paste the code again and change the import to com.mysql.* and then finally after that, still on the same line, I am going to write a declaration that this is an XML document. So open bracket, question mark, xml, space, version= and in quotes, the version is going to be 1.0, and then we will close it out with question mark and a greater than sign. So what we have done so far is just declare the classes that we want to import and use in Java and then outside of our Java code we have declared that this is an XML file. So let's go to the next line, and we are going to create the tours tag. So <tours>, and on the next line we will close that out with /tours tag. So we have an opening closing tag called to tours. And on the next line, inside of our tours tag, I'm going to write some Java code. So I will need to type a less than sign and a percent sign and Eclipse will autocreate the closing tag for me. And in here I'm going to initialize some variables that we will be using. So connection is the data type, and we will call this connection lowercase. We will just initialize it to null. We will do the same thing for a Statement. Statement is the class, statement lowercase is the name of the variable, set it equal to null, and the same thing for ResultSet. Capital R and capital S for the class name. And then result for the name, and again, we will initialize to null here. Now let's go down a few lines, and we need to connect to the database in a try catch block. So try and then some brackets and below the brackets, catch. The error is going to be of type SQLException. We will name it e. I will put the brackets there, and we will just type out .println and in the parentheses, pass in the string, "error connecting to database." No return to the try block. In there, we are going to create a new instance of the driver class by typing Class, capital C, .forName, and the name will pass in as a string so quotes and then pass in com.mysql.jdbc.Driver, with a capital D. And then after the close parenthesis type a dot and then newInstance. On the next line we will set the value of our connection. So connection all lowercase. We will set that equal to DriverManager, capital D, capital M, .getConnection, and then we pass in three parameters that are all strings. First the database url, then the username for the database and the password for the database. So in quotes url is going to be jdbc:mysql://localhost:3306. Remember that's the port for your MySQL server. And if you change it to something else in your settings, make sure you put that number here. And then /tours. That's the name of our database. After that string, type a comma. The next string is the username of our database. I kept mine at the default, which is root. And finally, the third parameter is also a string that's the password for the database, and I kept mine at the default as well, which is root. Make sure to add a semicolon after the end of the statement. And then here I am just going to type out.println, and in parentheses I will pass in the string "connected to database." So this is going to happen if there's no problem. So when we test this we should see either "connected to database" or "error connecting to database." So let's save the file and then test this using Tomcat. I am going to click OK to restart the server. And then in Eclipse's built-in browser I should see a message that says "connected to database." So that successfully connected to the database. If you have any problems with this, make sure that you're running both your MySQL server and Tomcat, and then check to see that the jar file is in the correct location so that you can load in the drivers. Finally, of course, make sure to check all of your code against mine, including the port number for your MySQL server.

Predicted Skills: Refactoring, Object oriented design, IntelliJ IDEA, Subversion, Standard widget toolkit, Facelets, Apache ant, Java architecture for XML binding, JSON, Maven, Java concurrency, Spring security, Log4j, Smart GWT, Spring boot, Libgdx, Text driven development, Play framework, Mockito, JavaFX, JavaServer Faces, Junit, Jboss seam, Eclipse RCP, Dropwizard, Zend framework, LAMP, OSGi, MyBatis, Netty, Guice, Winforms, Multithreading, Primefaces, Thymeleaf, PureMVC, Vaadin, Eclipselink, Aspect oriented programming, Wicket.

5.2 Example 2: A Music Course

This example shows video transcripts from a music course and corresponding predicted skills. Similarly, the order of the skills is based on the predicted probabilities.

Course Title: Drum Setup and Mic'ing in the Studio

Course Description: GRAMMY-winning recording engineer Ryan Hewitt explains his techniques for capturing drum sounds. "Drum Doctor" Ross Garfield consults with Ryan on a real-world studio setup for a session with A-list drummer Josh Freese.

Video Transcripts: There's a million ways to mic a drum kit. It's one of the most complicated instruments to record, 'cause there's so many things happening on it over a vast space in your studio. So what we're going to talk about right now is the way I mike drums for, you know, my typical rock, modern, pop kind of set up where you want a tight, punchy, controlled sound. So we're going to use a lot of close mics on each individual drum, we'll have a set of room mics that we'll play with, and then we'll have what I call my stunt mics, you know, the mics that are sort of the filler sound, if you will. Those are the ones I crush up, make them sound really nasty, and get a lot of character for the drum kit. One of the important things I think about with micing drums is what the mic is literally seeing. I mean, you know, hearing, but seeing. If you think about a microphone as having like a flashlight beam, you know, these are cardioid microphones here, a condenser and a dynamic, and they see like a flashlight. They're going to look in a particular direction. So if you take this guy, he's going to see, you know, something like this, as its concentrated sound source. It'll still pick up some things around the side, you know, 'cause the cardioid pattern is really like this, but the main thrust of it is going to be like a big flashlight beam. So when you think about where these are pointing, you'll know where the sound it's going to pick up is coming from. So it's sort of crucial to think about that when choosing microphones, and then putting them in front of the instrument that you are recording. Even though we're doing tight micing, we don't want the microphones to be so close to the source that they're picking up a very small portion of the drum, you know, or the drum head, the side of the drum. We want it to have a slightly more broad picture of that drum so that it's getting the maximum tonal quality and characteristic and volume of that drum, while balancing that with the spill, with the leakage that's coming from the other parts of the drum kits. So if you have a mic on the tom, you need to be careful of the bleed from the cymbals, and the bleed from the snare drum, stuff like that. Leakage is inevitable. We just want it to sound as good as possible. So that's where mic choice comes in, and aiming the mic properly at that particular drum. So the microphones that I'm going to use today are the microphones that are available at this studio. EastWest has a great selections of microphones, and I'm pretty lucky to have all the ones I like to use. That said, there are many, many, many mics you can substitute in all of these applications. We're going to use some really expensive stuff, but there are analogous microphones that are a lot less expensive that will sound very good. Is it going to sound as good as a FET 47? I don't know. Maybe not. But it's going to sound good for what you have. And it's important to remember that if you have a good microphone on a really good-sounding drum, it's going to sound good. This is my current favorite, the Beyer M 88. So I can start with this one, and if I'm not stoked on it I can try something else and move on to the next one. But as I say, if you have one microphone that's going to work on the kick drum, we'll make it work. You can follow some basic guidelines that we'll discuss, and find the sweet spot for that microphone to make it sound the best you can. Micing a drum kit is a difficult undertaking because of all the individual pieces. The important thing to remember is that all these pieces make up a whole. And we want to hear the drum kit as a whole. So we'll have the close mics, we'll have some far mics, we'll have the stunt mics, and we have to make them all work together. So with that in mind, let's mic up the drum kit.

Predicted Skills: R&B, Music remixing, Blues, Original music, Music supervision, MIDI, Studio recording, Audio mastering, Audio recording, Film scoring, Pop, Audio engineering, Lyrics, Music publishing, Professional audio, Sound reinforcement, Music production, Songwriting, A&R administration, Song production.