Connecting College Courses to Careers: An End-To-End Exploratory System

Nyssa Aragon
Chegg
Santa Clara, California
nyssa@chegg.com

Irina Borisova
Chegg
Santa Clara, California
irina@chegg.com

Shivani Gupta
Chegg
Santa Clara, California
sgupta@chegg.com

ABSTRACT

Job search and career discovery are well-known challenges for students: aligning academic achievements and interests with potential jobs requires external expertise from career offices and extensive research from students. Yet we know that job awareness among internship seekers remains quite low. We present an end-to-end exploratory system that allows a user to engage into career discovery and course planning with a single college course as an input. This experience is powered by three core machine learning models: a course sequence model that suggests other courses the user might have taken, a college course classification model that groups unique college courses into subject groups, and a course-to-skills and jobs model that projects these course categories into career domain. This paper discusses in detail the design and implementation requirements for the exploratory system along with the underlying models, their architecture and performance.

CCS CONCEPTS

• **Computing methodologies** → **Information extraction**; **Neural networks**; *Classification and regression trees*; • **Information systems** → *Expert systems.*

KEYWORDS

course classification, information extraction, course prediction

1 INTRODUCTION

Finding an internship and a first job has never been easy for students: a number of studies highlight difficulties that students face entering workforce (e.g. [15], [8], [14]). Many of these issues relate to the lack of information about the connection between the academic curriculum acquired in college and the types of relevant jobs. While there are several large-scale systems that attempt to bridge this gap through a manually or algorithmically designed crosswalk between majors and occupations (such as [10] and [9]), connecting specific coursework and professional roles at scale remains an open problem. This task is particularly important given the high numbers of students leaving college without a degree and struggling to find jobs utilizing their academic expertise and paying comparably to their incurred college debt [4], a diversity in major offerings and curriculum across them (e.g. [13], [12]), and an increasing value of internships for both employers and job applicants ([14], [3]).

We present a large-scale end-to-end exploratory system that allows a user to engage in career discovery and course planning with a single college course or a major as a minimal input. The scope of courses or majors is unlimited but expected to be within the U.S. educational system as the models that support the system are trained to represent the U.S. educational and professional landscape. Such system can support the following use cases:

(1) inform freshmen and sophomores with no declared majors about relevant internships given their coursework;
(2) help people who left college without a degree explore jobs relevant to their coursework;
(3) facilitate course selection.

Importantly, this exploratory system does not imply that a student taking a single course is qualified to do a recommended job; instead, it aims to raise student's awareness about the jobs that apply the skills usually learned through this course.

The rest of the paper is organized as follows: we present the main flow of the system and its modeling requirements and then discuss in detail each of the three core machine learning models that define it: a college course classification model that groups unique college courses into course categories, a course sequence model that suggests other courses a user might have taken, and a course category-to-skills and jobs model that projects these course categories into career domain.

2 ARCHITECTURE

2.1 System Flow

We describe an interaction flow in the course-to-job exploratory system following Figure 1. A session starts with a user entering her major or her college course name, her college name, and a year of enrollment. If a course name is provided, our first core model–a college course name classifier–maps the raw input to one of 2300 course categories, as described in 3. Majors are offered for user selection from the CIP list [10] or could be submitted as a typed-in input (then mapped to the CIP list through a proprietary raw major classifier). The college names are mapped to the IPEDS college classification [11] through a proprietary raw college name classifier.

Next, based on whether the input includes a major or a course, the system returns a list of courses that the user might have taken along or before her current college course or within her major program. Although available through two different inputs, the output is powered by one course sequence model, discussed in 4.2. Similar to the course classifier, the course sequence model operates on the course category level. To ensure that the user is familiar with the predicted course categories, we map them back to the college course offerings using the course classifier on a partner data set that covers U.S. college coursework, as introduced in 3.

The user is then asked to select the courses from the model predictions and add more, if none were relevant. If she types in additional courses, we apply the course classifier from the first

step to process the input. Finally, the system presents a list of job titles that are relevant given the selected coursework. This part is supported by the third core model - the course-to-jobs model.

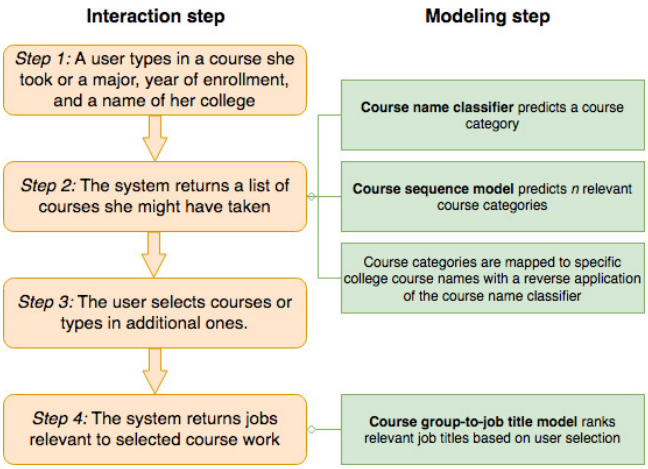

Figure 1: Course-to-jobs system workflow

2.2 Modeling Requirements

To support the described flow, our exploratory system has to satisfy the following design and modeling requirements:

- **Scalability**:to account for a large diversity of course and major offerings across the U.S. colleges, along with jobs, and to handle data sparsity, the system should utilize back-end models with canonical (or normalized) lists of majors, courses, colleges, and jobs. These canonical lists are either provided externally (the IPEDs classification of colleges and the CIP classification of majors) or developed internally (the job titles and skills). It is also critical for the core models to work with the same course units as their inputs – the course categories.
- **Coverage**: the system should support varied text inputs for colleges, majors and course names. This requires normalization models that would process raw inputs and map them into canonical sets of colleges, majors or course names. The underlying course and job data in model training should be representative of the U.S. colleges and labor market.
- **Familiarity**: the courses that it presents should be familiar to users. For courses, it means that the returned courses should be the ones offered in their college; for jobs, we expect to show the occupation titles that are common in the current labor market.
- **Relevance and transparency**: both intermediate and final output recommendations should be relevant to user's input. Each modeling component is evaluated on transparent metrics, and the system design supports step-by-step exploration and is highly interactive.

Although the scope of the modeling requirements for our system goes well beyond the course-focused models, we concentrate just on them for the sake of clarity and relevance.

Table 1: Classification examples for college course names

College Course Name	Predicted Course Category
Statistics for Health Pro	Medical Sciences: Basic: Biostatistics
Modern Arab-Muslim Thought	Political Science: Comparative Gov't: Gov't and Politics: Middle East
Folklore of Contemporary Greece	Classics: Greek Language and Literature

3 COURSE NORMALIZATION TASK: COLLEGE COURSE CLASSIFIER

The college course classifier plays a key normalization role in the course-to-job system: in the first step, it takes a raw course name and maps it to a course category; later it takes predicted course categories and maps them back to specific college course offerings. The classifier is described in detail in [2]. It is based on two core data sources: the first one is NPD PubTrack Digital, a proprietary textbook-to-course classification that includes over a hundred thousand ISBNs assigned to one of 2300 course categories. The course classification is hierarchical and includes 56 top-level course categories (e.g. *Engineering*) that further splits into granular categories, such as *Mechanical Engineering* and *Electrical Engineering*, and up to three levels deep to represent subject-specific subcategories). The second data set is Market Data Retrieval data (MDR) composed of 3.1M records of college courses taught in the U.S. institutions from 2005 to 2016. We use an extended overlap between the two data sets with 72,700 course names – course categories pairs to train two best-performing models: an LSTM classifier and a sequence-to-sequence (seq2seq) prediction model. The model architectures are described in detail in [2]. While the LSTM classifier performs best on the test set reaching an accuracy of 91% – two points above the seq2seq model, the latter does better in the human evaluation tasks reaching 88% relevance. Table 1 shows a few examples of the college course names and predictions made by the classifier.

The system implementation for raw input classification is based on these two models: we apply the seq2seq model to pre-populate a cache of the known college course names from the MDR data with their categories, and we use the LSTM model to power on-the-fly classification tasks. We search through the same cache, extended with the college and time information, to give the course categories in the course prediction task familiar names. For example, a course category of Art: Studio Art: Design: Graphic might have the following course offerings at the California State University - Fresno: GRAPHIC DESIGN: COMPUTER IMAGING, GRAPHIC CONCEPT DEVELOPMENT, BEGINNING 3D DIGITAL ART MODELING, INTRODUCTION TO COMPUTER ART.

4 COURSE PREDICTION TASK

4.1 ISBN-to-Course Category Mapping

While NPD PubTrack Digital has a wide coverage, it does not include all the ISBNs found in the Chegg orders. Since the order and

textbook data is instrumental for the course sequence and course-to-jobs models, we apply an auxiliary model that produces course category labels for the unclassified ISBNs. The model is based on a character-level fastText [1] embedding space built on the book titles and abstracts for the ISBNs with the course category labels. We retrieve a vector for an unassigned book, search for the nearest neighbor among the labeled books and propagate the label of the book with the highest score. This method reaches an accuracy of 90% and 86% for the first and second level course categories respectively and appears to be more accurate than the search for the nearest course category with the average course category embeddings.

4.2 Course Sequence Modeling

We aim to infer additional course enrollment information from incomplete data. Students come into our system providing limited input such as a major or partial coursework information. In order to recommend jobs on such limited input, we infer complete coursework. This is similar to the problem outlined in [6] with two differences. First, this problem requires the ability to infer courses from a major in addition to incomplete course information. Second, this model considers the sequence of courses taken and can infer courses already taken. This allows the job recommendations to be based on the student's current skill set.

Data – The coursework data focuses on mostly 4-year college curriculum. Vocational, graduate, and professional curriculum has little to no representation in the training data.

The training data is based on three sources. The first and largest is the partial courses data used in [6]. The second and the third are scraped course catalogs and course requirements for majors for a representative sample of 4-year U.S. colleges. From these sources, we obtain the course sequence and, for some sequences, the accompanied major.

In data sources where the course name was known, we identified the course categories for each college course using the model described in Section 3. In data sources where the ISBN was known, we mapped the book to an appropriate course category using the model described in 4.1.

Model – The prediction task is to choose the previous or next course taken based on either a course sequence, a major, or a course sequence and a major. The courses and majors are treated as bags-of-words. The input course sequence model was trained with a Keras [5] implementation of a GRU with a dropout on the input and recurrent state, an L2 regularization applied for learning the kernel weights, a tanh activation function, and an SGD optimizer with a learning rate of .001. An LSTM model was trained but required more computational resources with no lift in accuracy. The major bag-of-words vector was concatenated to the course GRU output. The last layers are a tahn activation layer and a softmax prediction layer. The loss is categorical cross-entropy.

Because people typically take multiple courses at once, there are multiple correct predictions for a given time step. However, when training with categorical cross-entropy, only one true class is given at a time. To accommodate for this, the same sequence may be used as input multiple times, once for each true class in the previous time step. An alternative to this model was tested – one that is trained on all the courses in the predicted time step at once, using binary cross-entropy and a sigmoid activation in the prediction layer. The advantage of this method is that it reflects the real-world multi-label nature of this problem. However, as shown on Table 2, the model with categorical cross-entropy loss achieved higher accuracy.

Table 2: Course sequence accuracy

Data	Cross-entropy	Course Accuracy	Level 1 Accuracy	Median Rank
Partial Courses Data	Binary	15%	42%	51
	Categorical	17%	44%	45
Scraped Courses	Binary	46%	76%	8
	Categorical	59%	80%	7
Scraped Major Requirements	Binary	22%	52%	22
	Categorical	28%	54%	18

Evaluation – Multiple accuracy metrics were used to evaluate the model. These metrics measure model performance against a withheld test set representative of the three data sources.

The first metric is the rate at which the actual course in the testing set is predicted. Because multiple courses are taken in a single time period and there are many correct answers, we look at the top-5 predicted courses as a more robust metric over the top-1 predicted course. The second metric takes advantage of the hierarchical structure of our course categories. We can measure the rate at which Level 1 course category, or the course subject, is accurately predicted. The third metric is the median rank of the actual course.

The metrics in Table 2 show that the model trained with categorical cross-entropy achieves higher accuracy of 59% and 80% for course and course subject on the scraped course catalog portion of the test set. This data contains many prerequisite course sequences which follow focused disciplines and time series. On the other hand, the data from partial courses reflects a subset of the coursework that people took. Some college curriculum includes general education requirements and elections from a diverse set of disciplines. Because one course may be completely unrelated to another course in a given sequence, predicting course sequences in this test set is a much more difficult task.

The evaluation metrics and model training optimize prediction accuracy, so that it tends to predict courses that are more likely to be included in a coursework given the input of the course sequence and major. This may not always align with other factors such as courses that are the most useful and relevant to career aspirations. To measure against these other factors, we would need additional evaluation data. To give a sense of the results, Table 3 shows predictions for a couple of course categories that were unseen during training.

5 JOB PROJECTION TASK: COURSE-TO-SKILLS AND JOB TITLES

This model aims at projecting course categories into career domain. We map course categories to canonical skills that we assume are

Table 3: Course sequence examples

Course Category	Previous Courses
Engineering: Mechanical: Conductive Heat Transfer	Engineering: Mechanical: Fluid Mechanics Engineering: General: Materials Science Engineering: Mechanical: Thermodynamics Engineering: Mechanical: Machine Design
Engineering: Mechanical: Plasticity	Engineering: Mechanics: Strength of Materials Physics: Introductory: College Engineering: Mechanical: Fluid Mechanics Mathematics: Calculus: Differential Equations

Table 4: Job title examples for course categories

Course Category	Relevant Jobs
Architecture: Urban Planning	City planner Urban designer Planning commissioner Water resources planner Ranger
Physics: Particle	Applied mathematician Research engineer Physicist Physics graduate student Postdoctoral research associate
Anthropology: Archeology: Introduction	Archeologist Assistant to the curator Historian Conservation technician Museum specialist

covered within the course curriculum and then find the relevant job titles for the given skills.

Data – We use partner electronic textbooks mapped to course categories as our primary data source.

Model – Course-to-skills and jobs model is powered by an in-house 300-dimensional word2vec [7] embedding space trained on 70 million resumes and representing professional and educational experience. To obtain this vector space, we treat each resume as a sequence of entities - majors, colleges, jobs, and skills - ordered chronologically. We apply a skip-gram model with hierarchical sampling, with a threshold of 100 observations for each entity, 10 negative samples drawn and a window covering 15 entities in a sequence. This choice of model and hyperparameters allows us to learn similarity relationships across different entities, such as finding relevant skills for a given title through nearest neighbor search). The embedding space is evaluated across several entity similarity tasks and downstream applications where the learned embeddings are used as an input for classification and clustering models.

Canonical lists of skills (18,000 skills) and job titles (8,500 job titles) are developed internally on top of a proprietary embedding model trained on the same resume data. The canonical skills include tools, or 'hard' skills (*java*, *salesforce*), topics (*particle physics*, *gene sequencing*), functional determinants (*project management*), activities (*greeting customers*) and 'soft' skills (*hard-working*). Both canonical lists are non-hierarchical.

Although a small sample of the resume data has course information, we chose against training on it due to a high variance in professional outcomes, a resume author pre-selection bias for courses (i.e. which courses they consider relevant to a desired profession) and a lack of conceptual connection between skills learned from a course and the resume jobs.

To learn an association between courses and skills, we start with extracting canonical skills from the e-book table of contents using n-gram matching and aggregating them for each course category. Next, we retrieve skill vectors from the resume embedding space to calculate a centroid for each course category using an embedding average weighted with skill-category counts. With this, we obtain course category representations that are reflective of the skill content observed in the categories' textbooks and embedded in the resume space. The latter allows us to perform nearest-neighbor search of the canonical job titles and rank them using cosine similarity. We also re-rank the observed skills via cosine similarity and additional heuristics.

Table 4 shows the top-ranked titles for a few course categories.

Evaluation – Due to the unsupervised nature of this model, finding meaningful evaluation metrics is not trivial. We made the following assumption for the quantitative evaluation: given a user textbook order data and her resume, the model should be able to predict the skills and job titles listed in that resume. This evaluation assumes that textbook orders are representative of the future career choices which, as we mentioned in 4.2, might be only somewhat true.

The evaluation set includes 100,000 textbook order-resume pairs for users who bought or searched for more than three books. We obtained course categories for the textbooks, extracted skills from their resumes using our in-house skill extraction model and compared these with the ranked skills of the corresponding course categories. We used exact string match and high cosine similarity (greater than 0.7 in order to bridge the gap between academic skills and practical skills) to compare the two lists. The average recall over all the users for top-20 skills was 12%. We also compared the first job in the resume with top-100 job titles relevant to the course categories. The recall percentage was 30% with an average rank of 15. Interestingly, the recall for jobs was higher than for the skills: it is indicative that the skills are self-declared and possibly reflect career aspirations and skill value assumptions held by the resume owners.

For the qualitative evaluation we asked human experts to review top-10 skills for 330 course categories and judge the quality of skills as relevant/irrelevant to their respective course categories. 77% of the course categories were found to have relevant top-ranked skills.

6 CONCLUSIONS

We present the course-to-jobs exploratory system and the key underlying models: trained to account for raw data, the models aim to process any course name, major and college input from a user and output relevant course and job suggestions. This allows our system to provide insights on professional outcomes at scale, with no college or discipline restrictions. As the system makes its way to the users, our next critical step is to collect significant user feedback from interactions with each of the system components and incorporate it into model development.

REFERENCES

[1] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching Word Vectors with Subword Information. *arXiv preprint arXiv:1607.04606* (2016).

[2] Irina Borisova. 2018. College Course Name Classification at Scale. In *Penstein Rosé C. et al. (eds) Artificial Intelligence in Education. AIED 2018. Lecture Notes in Computer Science.* Vol. 10948. Springer, Cham, 419–423. https://doi.org/10.1007/978-3-319-93846-2_78

[3] Gerard Callanan and Cynthia Benzing. 2004. Assessing the role of internships in the careerâĂŘoriented employment of graduating college students. *Education + Training* 46, 2 (2004), 82–89. https://doi.org/10.1108/00400910410525261 arXiv:https://doi.org/10.1108/00400910410525261

[4] Pew Research Center. 2014. The Rising Cost of Not Going to College. *Pew Research Center* (2 2014). https://www.pewsocialtrends.org/2014/02/11/the-rising-cost-of-not-going-to-college/

[5] François Chollet et al. 2015. Keras. https://keras.io.

[6] Jose Pablo González-Brenes and Ralph Edezhath. 2018. Inferring Course Enrollment from Partial Data. In *Penstein Rosé C. et al. (eds) Artificial Intelligence in Education. AIED 2018. Lecture Notes in Computer Science.* Vol. 10948. Springer, Cham, 429–432. https://doi.org/10.1007/978-3-319-93846-2_80

[7] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and Their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2 (NIPS'13).* Curran Associates Inc., USA, 3111–3119. http://dl.acm.org/citation.cfm?id=2999792.2999959

[8] Kerri A. Murphy, David L. Blustein, Amanda J. Bohlig, and Melissa G. Platt. 2010. The College-to-Career Transition: An Exploration of Emerging Adulthood. *Journal of Counseling & Development* 88, 2 (2010), 174–181. https://doi.org/10.1002/j.1556-6678.2010.tb00006.x arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/j.1556-6678.2010.tb00006.x

[9] University of California Berkeley Career Center. 2019. Connecting Majors to Careers. Retrieved May 15, 2019 from https://career.berkeley.edu/infolab/Majors2Careers

[10] U.S. Department of Education NCES. 2019. CIP 2010. Retrieved May 15, 2019 from https://nces.ed.gov/ipeds/cipcode/resources.aspx?y=55

[11] U.S. Department of Education NCES. 2019. Integrated Postsecondary Education Data System (IPEDS). Retrieved May 15, 2019 from https://nces.ed.gov/ipeds/use-the-data

[12] Baron Perlman and Lee I. McCann. 1999. The Structure of the Psychology Undergraduate Curriculum. *Teaching of Psychology* 26, 3 (1999), 171–176. https://doi.org/10.1207/S15328023TOP260302 arXiv:https://doi.org/10.1207/S15328023TOP260302

[13] Marie Petkus, John J. Perry, and Bruce K. Johnson. 2014. Core Requirements for the Economics Major. *The Journal of Economic Education* 45, 1 (2014), 56–62. https://doi.org/10.1080/00220485.2014.859961 arXiv:https://doi.org/10.1080/00220485.2014.859961

[14] Martha Ross, Kristin Anderson Moore, Kelly Murphy, Nicole Bateman, Alex DeMand, and Vanessa Sacks. 2008. *Pathways to High-Quality Jobs for Young Adults.* . Metropolitan Policy Program at Brookings - Child Trends.

[15] Nancy M. Wendlandt and Aaron B. Rochlen. 2008. Addressing the College-to-Work Transition: Implications for University Career Counselors. *Journal of Career Development* 35, 2 (2008), 151–165. https://doi.org/10.1177/0894845308325646 arXiv:https://doi.org/10.1177/0894845308325646