Interpretable embeddings to understand computing careers

Anonymous ACL submission

Abstract

We propose an approach for analyzing and comparing curricula of study programs in higher education. Pre-trained word embeddings are fine-tuned in a study program classification task, where each curriculum is represented by the names and content of its courses. By combining metric learning with a novel course-guided attention mechanism, our method obtains more accurate curriculum representations than strong baselines. Experiments on a new dataset containing curricula of computing programs demonstrate the interpretability power of our approach via attention weights, topic modeling, and embeddings visualizations. We also present a use case that compares computing study programs in the US and Latin America and showcase the capabilities of our method for identifying similarities and differences in topics of study in curricula from different countries.

1 Introduction

In recent years, the demand for computing careers has highly increased due to their influence in almost every area of human knowledge. In some countries, scientific associations such as ACM, IEEE, or ABET define guidelines to categorize computing careers through quality requirements (Shackelford et al., 2005). However, other countries do not have access to these specialized institutions or professionals, which may generate confusion among computing careers or mixtures between them (Sabin et al., 2016).

Prior work aims to analyze local context, and analyze market offer without following international standards. For example, de Alburquerque et al. (2010) and Prietch (2010) employ curriculums to analyze the market offer and propose to standardize Brazilian computing curriculums. More specifically, the Computer Science (CS) curriculum from the University of São Paulo is examined every year to incorporate current innovations (Macêdo, 2016).

In contrast to using local context, other works follow international standards. ICACIT (2019) elaborate an international accreditation process and make a suggestion to improve computing careers. Also, Murrugarra-Llerena et al. (2011) analyzes curriculums from Peru and Brazil using hierarchical clustering. They show groups identifying associations between both countries, and showing differences from the Peruvian context.

All previous works employ semi-automatic approaches and may need human intervention to reveal insights. Also, Murrugarra-Llerena et al. (2011) is the most related work to our approach, however, it only uses careers in Latin America and only course titles without descriptions. This representation may provide an incomplete story. We also believe hierarchical clustering may not provide explanations about what the model learns.

In this work, we hypothesize that fine-grained data sets and interpretable approaches are required to better understand computing curriculums. We collected a novel dataset combining course titles and their descriptions from US Universities. Then, we empower data analysts with an interpretable model combining course-guided attention and metric learning. Our approach identifies core courses per computing career.

Using our collected dataset, we compare our course-based attention approach to traditional text embedding techniques, fine-tuned Bert models, and metric learning approaches. Our approach outperforms all of them. We also show qualitative results via attention weights, topic modeling, and embedding visualizations. These results highlight the interpretability of our approach by identifying relevant words for each computing career. Finally, we develop an application to visualize how Latin America computing programs are identified with international ABET computing careers.

In summary, our main contributions are:

- A novel dataset of US computing careers, with

representative Latin America universities.

- An examination of attention, metric learning, and Bert modules to generate more interpretable embedding representations.
- An application to categorize a computing curriculum compared to international standards.

2 Approach

Since no prior dataset exists with course titles and their descriptors, we first collect such dataset in section 2.1. Next, we describe our course-based attention approach in section 2.2 and conclude with our implementation details in section 2.3.

2.1 Dataset

We collected 300 curriculums among computing programs from the best universities in USA. As a quality criterion, we follow ABET program[1] to filter them. In detail, we collected 100 Computer Science (CS), 100 Computer Engineering (CE), 38 Information Technology (IT), 34 Information Science (IS), and 28 Software Engineering (SE) curriculums. Each curriculum consists of a set of courses including their title and course description. We depicted an example from each computing program in Table 2 (appendix).

In addition to the USA curriculums, we collected 18 LATAM curriculums to analyze their degree of internationalization. We prioritize high-ranking universities that claim a CS profile and are freely available. Our selected curriculums come from Brazil, Colombia, Mexico, and Peru. We translate them to English to avoid multi-lingual issues.

We used web scrapping with beautiful soup library[2]. For some curriculums, we require a manual inspection (e.g. to remove information about credits and hours) to ensure a uniform structure. We plan to release the data set after article acceptance.

2.2 Course-based attention approach

Our course-based attention approach $Bert_{met+att}$ aims to learn the importance of each course associated with computing careers. As shown in Figure 1, our approach receives a computing curriculum composed of courses and their $Bert$ embeddings $curriculum_{emb}$. Then, we compute $att_{weights}$ of each course. Using these weights, we calculated a weighted average over the courses and generate a new curriculum embedding $curriculum_{emb\_avg}$. Finally, we collapsed the generated embedding in 100 features.

To learn well-defined groups among computing careers, we employ metric learning with the following triplet loss (Schroff et al., 2015), where N is the number of instances in a batch, α is the triplet margin with value 0.3[3], and θ denotes the learnt parameters.

$$L(c;\theta) = \sum_{i=1}^{N} \Big[\frac{(c_i^a . c_i^p)}{||c_i^a|| \times ||c_i^p||} - \frac{(c_i^a . c_i^n)}{||c_i^a|| \times ||c_i^n||} + \alpha \Big]_+$$

Given an anchor curriculum (c_i^a) and using instances in the same batch, we select curriculums with the same category as positive annotations (c_i^p), and curriculums from different categories as negative annotations (c_i^n). Data points were randomly sampled with a batch size of 64 to ensure that every category is present in each iteration.

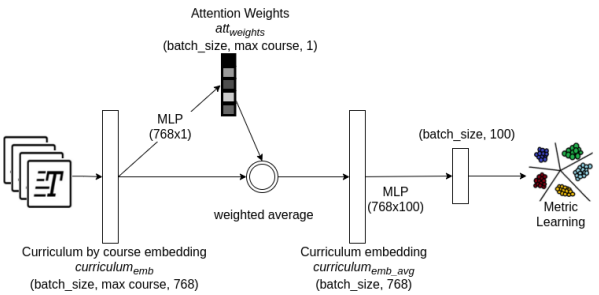


Figure 1: Our course-based attention approach. It generates an interpretable representation of curriculums via attention weights and metric learning. Attention highlights core courses, while metric learning learns boundaries to form well-defined groups. Both components are crucial to find accurate representations.

2.3 Implementation details

We implemented the networks using Pytorch (Paszke et al., 2019) and metric learning library (Musgrave et al., 2020). We ran each experiment ten times with different seeds using SGD. From preliminary experiments, we vary the batch size in the range $[32, 64, 128]$ and the embedding output in the range $[128, 256, 512]$. Then, we select the best configuration ($batch\ size = 64$ and $embedding\ size = 128$) in our validation set.

3 Experimental validation

3.1 Baselines

We compare three traditional methods: $Word2vec$ (Mikolov et al., 2013), $Glove$ (Pennington et al., 2014), and $Bert$ (Devlin et al.,

[1] https://www.abet.org/

[2] https://www.crummy.com/software/BeautifulSoup/

[3] Default parameter suggested by metric learning library.

2019). Additionally, we fine-tuned Bert's embedding with our training data:

- $Bert_{unsup}$, unsupervised finetuning using language modeling.
- $Bert_{sup}$, supervised finetuning using classification labels.

We also consider metric learning baselines:

- $Bert_{met}$, supervised metric learning using $Bert$.
- $Fusion_{met+att}$, supervised metric learning with attention over $Glove$, $Word2vec$ and $Bert$.

3.2 Evaluation protocol

From our US dataset, we split our data in 60% for training, 20% for validation, and 20% for testing. Using the train set, in non-pretrained models, we learn a new embedding representation. Then, we use those embeddings to feed machine learning classifiers. To select the best parameter configuration, each classifier was evaluated on a validation set and the configuration with higher F1was selected for testing. For all non-pretrained models, we trained them with ten different seeds and report their F1 average.

3.3 Quantitative experiments

We aim to validate which approach generates a more precise representation for classification. From the computed embeddings, we trained four classifiers: K-nearest neighbour (KNN), Linear Regression (LR), Linear Support Vector Machine (LSVM), and Radial Support Vector Machine (RSVM) with a proper search range of parameters (detailed in Section A.2).

Table 1 shows F1-score for all the embeddings in the test set. We observe that $Bert_{met+att}$ outperforms on average all other competitors and boosts the RSVM classifier. $Fusion_{met+att}$ is the second-best performer and reports competitive results with the KNN and RSVM classifiers. From pre-trained embeddings, the best baseline is $Bert$ and presents the best result in LR and LSVM. On the other hand, the weakest baselines are $Glove$ and $Word2vec$. A possible explanation is the low number of features and scarce training data.

Models	KNN	LR	LSVM	RSVM	Avg
$Word2vec$	55.10	71.00	57.10	55.90	59.77
$Glove$	54.90	73.10	64.80	64.80	64.40
$Bert$	48.50	**80.30**	**75.90**	65.90	67.65
$Bert_{sup}$	55.00	78.10	71.40	68.20	68.17
$Bert_{unsup}$	64.20	73.00	69.50	70.10	69.20
$Bert_{met}$	**73.40**	72.60	72.10	72.80	72.72
$Bert_{met+att}$	71.60	75.60	75.70	**75.60**	**74.82**
$Fusion_{met+att}$	72.40	69.60	74.00	75.40	72.85

Table 1: F1-score results on the test set of our embeddings with KNN, LR, LSVM and RSVM classifiers. Last three baselines used metric learning.

To conclude, we believe our improved performance is due to our interpretable embedding via attention weights and metric learning modules as detailed in the section below.

3.4 Qualitative experiments

3.4.1 Attention weights

To analyze the internal functionality of our approach, from each curriculum, we extract the attention weights of each course. Then, we rank them in decreasing order and select the top five. We group these selected courses per computing program and create a word cloud visualization.

Figure 2 shows these computed word clouds for each computing program. We find that words with a higher number of occurrences are relevant to their respective category name. We observe that "computer" is common among all computing careers, but it is more relevant for CS, CE, and SE; while it has less importance for IT and IS.

CS suggests a strong association to algorithms and computer; CE to design and computer. IT to Information Management and System; IS to Principles and Information Database and SE to Systems and Programming. All these associations confirm the identity of each career, and we observe that IT and IS highlight information-related courses, while CS, CE, and SE are more technical. For example, CS focuses on algorithm efficiency, CE specializes in hardware design, and SE promotes programming skills in general. The frequencies of each word by category are in Section A.3.2.

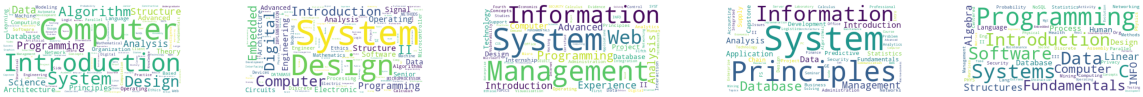
Also, we selected words with highest attention, and identify topics using (Popa and Rebedea, 2021)[4] in Section A.3.1. Similarly, we observe key differences among careers.

3.4.2 Embedding visualizations

To understand if our attention-guided interpretability is able to generate meaningful embeddings, we visualize $Bert$ and $Bert_{met+att}$ through Umap (McInnes et al., 2018) in Figure 3.

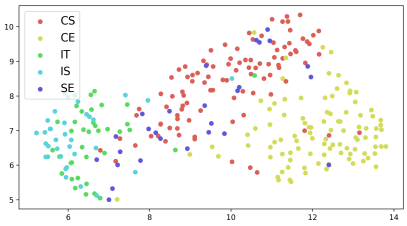
$Bert_{met+att}$ separates more clearly computing programs than $Bert$. CE and CS show well-defined boundaries rather than in $Bert$ Umap, and overlap is minimized among all categories. We also

[4]https://huggingface.co/cristian-popa/bart-tl-all

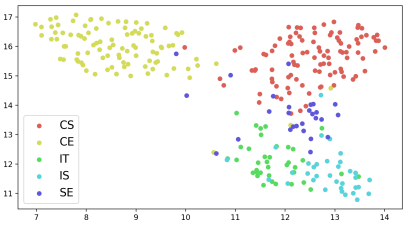


(a) CS (b) CE (c) IT (d) IS (e) SE

Figure 2: Word Clouds from courses of top 5 attention weights obtained with $Bert_{met+att}$ model on the test set with (a) Computer Science (CS), (b) Computer Engineering (CE), (c) Information Technology (IT), (d) Information System (IS), and (e) Software Engineering (SE).



(a) $Bert$



(b) $Bert_{met+att}$

Figure 3: Umap visualizations for (a) $Bert$ and (b) $Bert_{met+att}$. $Bert_{met+att}$ embeddings distinguishes better each computing category, while $Bert$ present some overlaps.

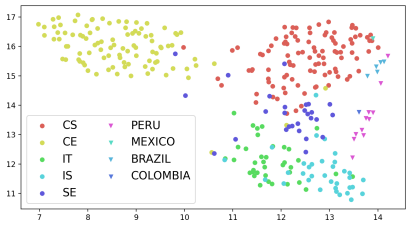
observe that IT and IS are close by. A possible explanation is through their shared financial and administration components. On the other hand, SE seems to be hard to form its own group. Apparently, it has pieces of all careers. We attribute this finding due that SE is a new career, less well-established.

Finally, we also analyze the attention weights of our best competitor $Fusion_{met+att}$ in Section A.3.3. $Bert$ is the most important representation, which confirms our choice of $Bert$ embedding.

4 Application: internationalization

For our application, we investigate how LATAM computing careers relate to international standards. We used our learnt $Bert_{met+att}$ to project unseen CS LATAM computing careers and relate them with US standards in Figure 4 using Umap.

LATAM curriculums (in triangles) form two predominant groups: one near to the CS group, and



(a) $Bert_{met+att}$ *with LATAM countries*

Figure 4: Umap visualizations for $Bert_{met+att}$ with LATAM countries. Triangles represent LATAM countries, conformed by Peru, Mexico, Brazil, and Colombia. These countries form two groups: one near CS and other near IT, IS, and SE.

the other group surrounding IS and SE. Also, no LATAM career surrounds the CE profile. From this visualization, we infer that LATAM careers are different from the US because none of them lay inside US groups. Then, we perform a closer study on individual LATAM countries. Brazil and Mexico have a clear CS profile. Also, Mexico seems much more integrated with the US profile maybe because of its near geographic location. On the other hand, Peru has a mixed profile between CS, SE, and IS; which may suggest a better definition of courses per career. Finally, Colombia belongs to SE.

5 Conclusion

In this article, we explore an interpretable way to classify computing curriculums combining course-guided attention and metric learning. Our approach finds more cohesive groups with clear separation among them. These groupings are helpful for different machine learning models. Also, we analyze what our approach learns via attention weights, topic modeling, and visualization techniques.

As future work, we plan to evaluate our approach in other NLP domains. Also, we will combine $Bert_{met+att}$ and $Fusion_{met+att}$ creating a new embedding combining course-guided attention, a embedding-guided attention ($Bert$, $Glove$, and $Word2vec$), and metric learning.

References

Pereira L. Z. de Alburquerque et al. 2010. Uma Análise da Oferta e Abordagem Curricular dos Cursos de Bachelorado em Sistemas de Informação no Brasil. *Workshop sobre Educação em Computação (WEI)*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (ACL)*.

ICACIT. 2019. Accreditation process. urlhttp://www.icacit.org.pe/web/acreditacion/sobre-acreditacion/ciclo-de-acreditacion.html.

Batista Daniel Macêdo. 2016. Nova grade curricular do BCC-IME-USP. *Workshop sobre Educação em Computação (WEI)*.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Grossberger. 2018. Umap: Uniform manifold approximation and projection. *The Journal of Open Source Software (JOSS)*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *International Conference on Learning Representations (ICLR)*.

Nils Murrugarra-Llerena, Fernando Alva-Manchego, and Solange Oliveira. 2011. Comparação de Grades Curriculares de Cursos de Computação Baseada em Agrupamento Hierarquico de Textos. *Workshop sobre Educação em Computação (WEI)*.

Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. 2020. Pytorch metric learning.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Neural Information Processing (NIPS)*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Cristian Popa and Traian Rebedea. 2021. BART-TL: Weakly-supervised topic label generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

S. S. Prietch. 2010. Mapeamento de Cursos de Licenciatura em Computação seguido de Proposta de Padronização de Matriz Curricular. *Workshop sobre Educação em Computação (WEI)*.

Mihaela Sabin et al. 2016. Latin American Perspectives to Internationalize Undergraduate Information Technology Education. *Working Group Report*.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Russell Shackelford et al. 2005. Computing curricula 2005. Technical report, ACM, IEEE and AIS.

A Appendix

We provide a sample curriculum per computing career and additional details for quantitative and qualitative experiments. For quantitative, we provide details about parameter ranges for model selection. For qualitative experiments, we provide results on topic modeling, show counts for attended courses from our attention module, and attention weights for the best baseline competitor.

A.1 Sample curriculum

Each collected curriculum in our dataset consists of a set of courses. Each course has an associated title and description. We depicted an example from each computing program in Table 2.

A.2 Range parameters for experiments

We mention the employed machine learning models with their associated parameter values below:

- For k-nearest neighbour (KNN), we evaluate k with values [3,5,7].
- For Linear Regresion (LR), we evaluate cost C with values $[2^-5, 2^-3, 2^-1, 2^1, ..., 2^{15}]$.
- For Linear SVM (LSVM), we evaluate cost C with values $[2^-5, 2^-3, 2^-1, 2^1, ..., 2^{15}]$.
- For Radial SVM (RSVM), we evaluate cost C with values $[2^{-5}, 2^{-3}, 2^{-1}, 2^1, ..., 2^{15}]$, and gamma with values $[2^{-15}, 2^{-13}, 2^{-11}, ..., 2^1, 2^3]$.

A.3 Quantitative experiments

A.3.1 Topic modeling

As a complementary way to understand our selected courses, we selected the ten words with the highest attention, and input them to BART topic model (Popa and Rebedea, 2021)[5] to name them.

[5]https://huggingface.co/cristian-popa/bart-tl-all

Career	Course title	Description
CS	Algorithms and Data Structures	Study of data structures and algorithms ...
CE	Computer Architecture and Design	Principles of RISC-type CPU instruction set and ...
IT	Information Technology Security	Information technology security from a manager ...
IS	Information Systems Applications	Concepts and production skills ...
SE	Software Engineering Design	Techniques and methodologies ...

Table 2: Sample curriculum showing course titles and their description per computing career.

The named topics are shown in Table 3. CS, CE, and IT share the word computer highlighting the importance of computing fundamentals, while IT and IS share the topics management and information relating to business knowledge. Also, programming skills are shared among CS and SE careers.

Career	Topic
CS	computer programming data
CE	computer design system
IT	management information computer
IS	system management information data
SE	software programming language

Table 3: Topic identified with each computing career using BART (Popa and Rebedea, 2021) model.

A.3.2 Counts for attended courses

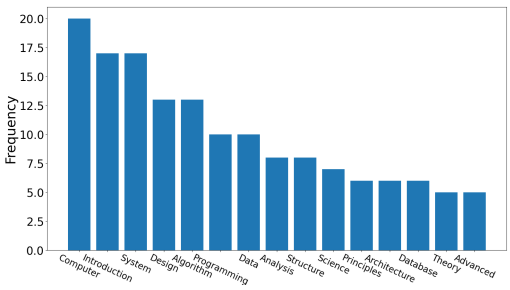
Figure 5 shows the frequency of the top fifteen courses per category in decreasing order. We find the following associations per each computing career:

- CS highlights computer, introduction, system, design, algorithm, programming, and data courses.
- CE focuses on systems, design, computer, digital, and embedded.
- IT has relevant terms such as system, management, information, web, and programming.
- IS focuses on systems, principles, information, database, and management.
- SE highlights programming, systems, data, introduction, C, and software.

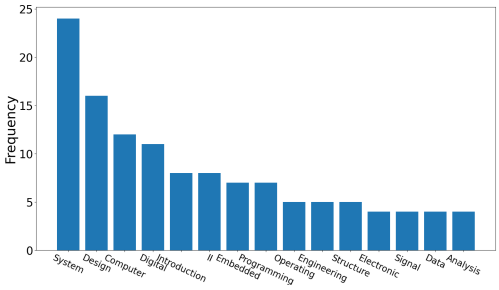
In summary, IT and IS are related to management and information knowledge. CE focuses on hardware concepts such as systems, design, and digital. Finally, CS and SE focus on software development related to programming, data, and algorithm courses.

A.3.3 Attention weights best competitor

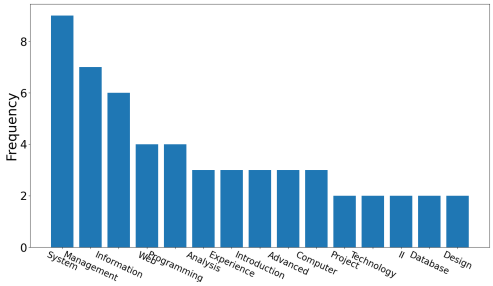
We analyze our best competitor $Fusion_{met+att}$ to discover interesting knowledge. We extract attention weights for each embedding representation. On average we obtained 0.2149 weight for $Glove$, 0.0621 for $Word2vec$, and 0.7230 for $Bert$. This finding confirms our election to select $Bert$ embedding in our approach. Also, it is interesting to see that $Glove$ and $Word2vec$ also have complementary and meaningful knowledge for better representation. Probably $Word2vec$ and $Glove$ provide local information to the $Bert$ embedding. Note, that their attention scores have the same order as their correspondent F1-score (see three first rows in Table 1).



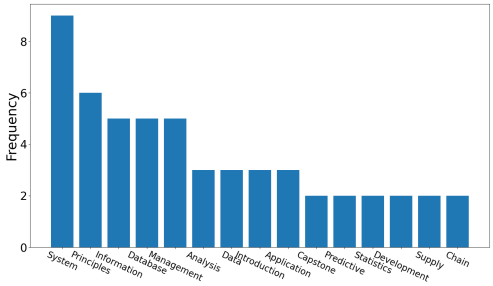
(a) CS



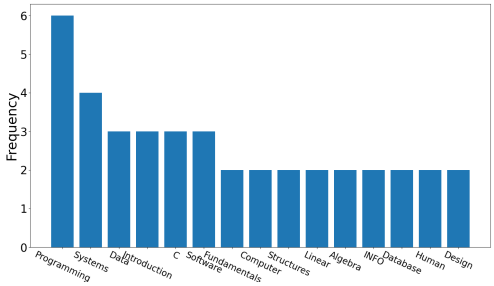
(b) CE



(c) IT



(d) IS



(e) SE

Figure 5: Top fifteen frequency terms of each category. The X-axis shows the word term, while Y-axis shows their frequency. The categories are (a) Computer Science (CS), (b) Computer Engineering (CE), (c) Information Technology (IT), (d) Information System (IS), and (e) Software Engineering (SE).