Improving Embeddings Representations for Comparing Higher Education Curricula: A Use Case in Computing

Jeffri Murrugarra-Llerena[1], **Fernando Alva-Manchego**[2], **Nils Murrugarra-Llerena**[3]
[1]Institute of Informatics, Federal University of Rio Grande do Sul (UFRGS), Brasil
[2]School of Computer Science and Informatics, Cardiff University, UK
[3]School of Computing, Weber State University, USA
jeffri.mllerena@inf.ufrgs.br, alvamanchegof@cardiff.ac.uk,
nmurrugarrallerena@weber.edu

Abstract

We propose an approach for 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 with curricula of computing programs demonstrate the intuitive power of our approach via attention weights, topic modeling, and embeddings visualizations. We also present a use case comparing computing curricula from USA and Latin America to showcase the capabilities of our improved embeddings representations. Our code and data are available in https://github.com/Artcs1/DL_curriculas.

1 Introduction

Several stakeholders in Education compare and assess curricula, such as Governments aiming to improve the competitiveness of their local programs (Adamson and Morris, 2007), or academics looking to analyze market offer and suggest new courses and innovations (Prietch, 2010; de Alburquerque et al., 2010; Macêdo, 2016). Despite their benefits, manual comparisons can be time-consuming, and are prone to biases from human assessors due to their personal beliefs and perspectives (Kawintiranon et al., 2016). This situation becomes more challenging when studying curricula related to programs in computing due to their fast-paced evolution, which may prevent suitable analysis and evaluation by human intervention (Föll and Thiesse, 2021). To aid higher education stakeholders, we propose a method to automatically compare curricula based on the names and content of their courses, and test it on study programs in computing.

Previous work has represented curricula using bag-of-words, and relied on clustering algorithms to identify groups of commonly-studied topics in data science (West, 2017), or to find links between study programs in different countries and across computing disciplines (Murrugarra-Llerena et al., 2011). Others have attempted to associate curricula to knowledge areas defined by international associations, such as ACM or IEEE (Shackelford et al., 2005), using standard TF-IDF representations (Kawintiranon et al., 2016) or topic modelling (Matsuda et al., 2018; Föll and Thiesse, 2021).

In this work, we follow the current trend in NLP applications and use pre-trained word embeddings, such as BERT (Devlin et al., 2019), to obtain better representations of textual curricula. We fine-tune these embeddings on a computing discipline classification task, using a newly-collected dataset that includes study programs from accredited universities in the USA and Latin America. We introduce a course-guided attention mechanism that allows the model to identify which courses are more relevant for each computing discipline, and pair it with metric learning (Xing et al., 2002) to learn well-defined groups. Experiments with different types of pre-trained word embeddings and classification algorithms, show that our proposed approach generates an accurate representation of computing curricula that allows human understanding. We also show qualitative results via attention weights, topic modeling, and embedding visualizations. These results highlight the benefits of our approach for identifying relevant words for each computing curriculum. Finally, we exploit these embeddings to visualize how Latin America computing programs relate to recognized ABET disciplines.

In summary, our main contributions are:

- A novel dataset of US computing curricula and relevant programs from Latin America .
- An examination of attention, metric learning, and BERT modules to generate more intuitive embedding representations.
- An application that compares a computing

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 11299–11307
December 7-11, 2022

curriculum to international standards.

2 Computing Curricula Dataset

We collected curricula from university study programs from different countries and categorized them into five computing disciplines: Computer Science (CS), Computer Engineering (CE), Information Technology (IT), Information Science (IS), and Software Engineering (SE).

Each curriculum consists of a set of courses including their title and course description, as depicted in Table 4 in Appendix A.1. The dataset consists of two sections:

- **USA.** Contains 296 curricula from US universities in the top 1000 of the QS rankings from 2021,[1]which were also accredited by ABET.[2]
- **LATAM.** Contains 18 curricula from highly-ranked universities in Brazil, Colombia, Mexico, and Peru.[3] These study programs claim to correspond to the Computer Science discipline. These curricula were first translated using Google Translate, and then manually revised by an author that is fluent in English and Spanish to resolve inconsistency issues such as mistranslated words, wrong word order, etc.

The USA portion of the dataset is used to train and evaluate embeddings representation in a computing discipline classification task (Sec. 5), while the LATAM portion is exploited to analyse the degree of internationalization of Latin American study programs in computing (Sec. 7).

2.1 Dataset Statistics

We compute statistics of our dataset such as number of curricula, and average number of courses per curriculum category with their standard deviations in Table 1. Also, as a way of visualization, we create a word cloud of all our dataset using course titles and their descriptions in Figure 1. We observe that many courses are introductory and applied since words such as *introduction* and *application* are more frequent. On the other hand, core courses of computing categories cannot be easily found, which ensures that our dataset is not biased to a specific class regardless of it being imbalanced.

[1]https://www.topuniversities.com/university-rankings/world-university-rankings/2022

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

[3]Scored by Google search.

	USA		LATAM	
Career	#Curr.	Avg. #courses	#Curr.	Avg. #courses
CS	100	48.38±25.82	18	69.00 ± 18.90
CE	98	53.71±22.10	-	-
IT	37	43.10±16.91	-	-
IS	34	40.38±15.60	-	-
SE	27	46.25±13.62	-	-
Total	296	49.67±33.69	18	69.00 ± 18.90

Table 1: Statistics of our dataset. We present number of curricula and average number of courses per curriculum category and their associated standard deviations.

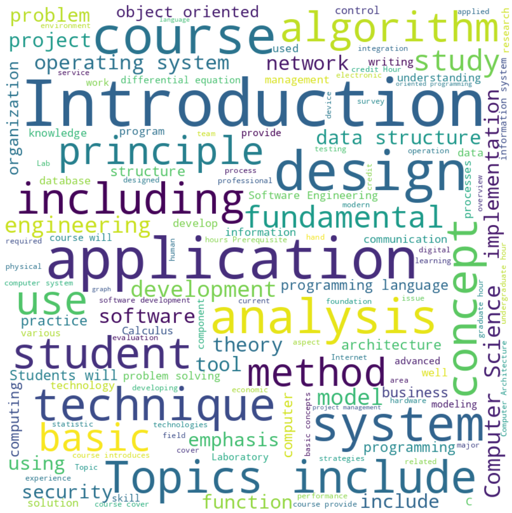


Figure 1: Word cloud of our dataset.

3 Approach

In order to obtain better representations of textual curricula, we propose to use pre-trained BERT (Devlin et al., 2019) embeddings that have been fine-tuned in a computing discipline classification task, using an approach that combines a novel course-based attention mechanism and metric learning. Figure 2 shows an overview of our method. Course-based attention identifies the most and the least important courses following the intuition of core and elective courses, while metric learning learns boundaries to form well-defined groups.

Course-Based Attention. Our course-based attention approach aims to learn the relevance of each course associated with computing curricula. As shown in Figure 2, it receives a computing curriculum composed of courses and their $Bert$ embeddings $curriculum_{emb}$. Then, we compute $att_{weights}$ for each course. Using these weights, we calculate a weighted average over the courses and generate a new curriculum embedding $curriculum_{emb\_avg}$. Finally, we collapse the gen-

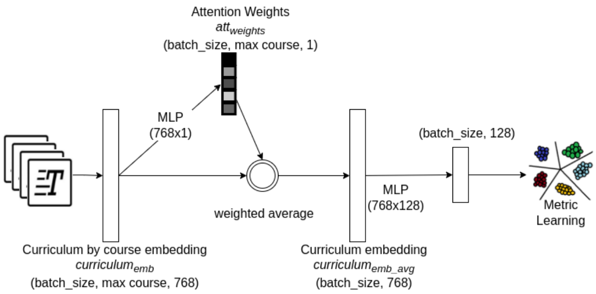


Figure 2: Our course-based attention approach. It generates an intuitive representation of curriculums via attention weights and metric learning. Attention highlights core courses, while metric learning learns boundaries to form well-defined groups. Both components are crucial to find accurate representations.

erated embedding into 128 features.

Metric Learning. Using these features, we learn well-defined groups among computing curricula. 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,[4] and θ denotes the learnt parameters.

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

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 curricula 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.

3.1 Implementation Details

We implemented the networks using Pytorch (Paszke et al., 2019) and the metric learning library (Musgrave et al., 2020) on a RTX 3060 GPU. 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.

4 Experimental Setting

4.1 Baselines

We compare two traditional baselines: Majority and Random; and five traditional methods for obtaining text representations: Topic Modeling using BERT topic (Grootendorst, 2022), TF-IDF (Sammut and Webb, 2010), $Word2vec$ (Mikolov et al., 2013), $Glove$ (Pennington et al., 2014), and $BERT$ (Devlin et al., 2019)[5]. For Topic Modeling, we select five topics and match them with our computing careers. Additionally, we fine-tuned BERT embeddings with our training data in two ways:

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

We also consider metric learning competitors:

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

4.2 Evaluation Protocol

From the USA dataset, we split the data into train (60%), validation (20%), and test (20%) in a stratified fashion. We use the train set to learn embedding representations using our proposed approach. Then, we use those embeddings to train several machine learning classifiers. To select the best parameter configuration, each classifier was evaluated on a validation set and the configuration with the highest F1 was selected for testing. For all non-pretrained models, we trained them with ten different seeds and report their average F1.

5 Quantitative Experiments

We aim to validate which approach generates a more precise representation for classification and human intuition. Using the newly-computed embeddings, we trained four classifiers: K-nearest neighbour (KNN), Logistic Regression (LR), Linear Support Vector Machine (LSVM), and Radial Support Vector Machine (RSVM) with a proper search range of parameters (detailed in Appendix A.2.1).

Table 2 shows F1-scores for all classifiers trained with different types of curricula representations, with exception of the first three lines. For each approach, embedding sizes are reported. We observe that $Bert_{met+att}$ outperforms, on average, all other competitors and boosts the RSVM classifier. This

[4] Default parameter suggested by the metric learning library.

[5] Course number of words do not exceed the maximum length token of BERT

Embedding Type	KNN	LR	LSVM	RSVM	Avg	Emb.size
$Majority$	-	-	-	-	10.00	-
$Random$	-	-	-	-	18.38	-
$Topic\_Modeling$	-	-	-	-	39.80	-
$TF-IDF$	63.10	67.70	63.40	58.00	63.05	20k
$Word2vec$	55.10	71.00	57.10	55.90	59.77	200
$Glove$	54.90	73.10	64.80	64.80	64.40	200
$Bert$	48.50	**80.30**	**75.90**	65.90	67.65	768
$Bert_{sup}$	55.00	78.10	71.40	68.20	68.17	768
$Bert_{unsup}$	64.20	73.00	69.50	70.10	69.20	768
$Bert_{met}$	**73.40**	72.60	72.10	72.80	72.72	128
$Bert_{met+att}$	71.60	75.60	75.70	**75.60**	**74.82**	128
$Fusion_{met+att}$	72.40	69.60	74.00	75.40	72.85	128

Table 2: F1-score results on the test set of our embeddings with KNN, LR, LSVM and RSVM classifiers. First seven baselines are traditional methods, intermediate two baselines are finetuned BERTs, and last three baselines used metric learning.

suggests that the generated embeddings better differentiate computing curricula, and can be helpful for visualization tasks (See Sec. 6.1). On the other hand, $Fusion_{met+att}$ 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. The weakest baselines are $Majority$, $Random$ and $Topic\_modeling$ which highlight that these approaches are not appropriate for this task.To conclude, we believe our improved performance is due to our intuitive embedding via attention weights and metric learning modules.

6 Qualitative Experiments

6.1 Embedding Visualizations

To understand how meaningful the generated embeddings are , we visualize $Bert$ and $Bert_{met+att}$ through Umap (McInnes et al., 2018) in Fig. 3.

$Bert_{met+att}$ separates computing programs more clearly than $Bert$. CE and CS show more well-defined boundaries than in $Bert$'s Umap, and overlap is minimized among all categories. We also observe that IT and IS are close to each other. A possible explanation is through their shared financial and administration courses. On the other hand, SE seems to be unable to form its own group. Apparently, it has pieces of all disciplines. We attribute this finding to the fact that SE is a new career, less well-established. Finally, we also analyze the attention weights of our best competitor $Fusion_{met+att}$ in Appendix A.3.4.$Bert$ is the most important representation, which confirms our choice of $Bert$ embedding.

6.2 Attention Weights

To verify that our model identifies core courses per discipline, we extract the attention weights of each computing course from the $Bert_{met+att}$ model. 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 using course titles.

Figure 4 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.

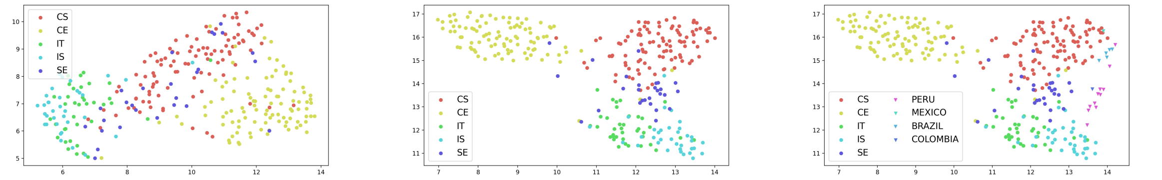
CS suggests a strong association to *algorithms and computer*; CE to design and computer. IT to *Information Management and System*; IS to *Principles and Information Database* and SE to *Systems and Programming*. All these associations confirm the identity of each career, and we observe that IT and 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 Appendix A.3.2, while a comparative analysis of word frequency with TF-IDF is shown in Appendix A.3.3. In addition, we selected the most frequent course titles, and identified topics using (Popa and Rebedea, 2021)[6] in Appendix A.3.1.

7 Application: Internationalization

As a use case, we investigated how LATAM computing careers relate to international standards. We used $Bert_{met+att}$ to project unseen CS LATAM computing curricula and try to relate them to USA standards in Figure 3 (c) using Umap.

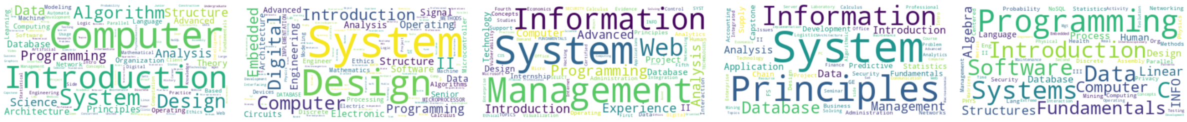
LATAM curricula (in triangles) form two groups: one near CS, and one surrounding IS and SE. Also, no LATAM curriculum is close to the CE profile. From this visualization, we infer that LATAM curricula are different from the US because none of them lay inside US groups. Then, we perform a closer study on individual LATAM countries. Brazil and Mexico have a clear CS profile. Also, Mexico seems much more integrated with the US

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



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

Figure 3: Umap visualizations for (a) $Bert$ and (b) $Bert_{met+att}$. $Bert_{met+att}$ embeddings better distinguish each computing category, while $Bert$ presents some overlaps. (c) Umap visualizations for $Bert_{met+att}$ with LATAM countries. Triangles represent LATAM countries. These countries form two groups: one near CS and other near IT, IS, and SE.



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

Figure 4: 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).

profile. 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.

8 Discussion

External validation: LATAM careers are different from US ones. To increase reliability of our results, we could ask career directors from LATAM Universities for external validation. However, most of them are busy professionals who may not be available. As an alternative, we support our results on previous work. Araujo et al. (2020); Takada et al. (2020); Sabin et al. (2016); Cuadros-Vargas et al. (2013) show that most LATAM universities are different between them (even with similar goals). They do not follow common standards, with discrepancies in multiple curricula. These differences do not tend to occur on USA curricula, which appears to follow international standards.

Reducing subjectivity in curricula comparison. Our approach helps reduce human bias by allowing an automatic comparison of a curriculum to international standards. Discovering relevant courses via attention weights is particularly useful. For example, Table 3 shows the top-5 most relevant courses, according to our attention module, from three programs (two from CS and one from SE). For CS, we observe overlaps for design of algorithms, data structures, and computer systems (highlighted in light gray). In contrast, the overlap with the SE program is limited to analysis of algorithms.

CS01	CS02	SE01
Algorithms and Complexity Design and analysis	Design and Analysis of Algorithms	Design and Analysis of Algorithms
Data Structures and Object-Oriented Design	Data Structures	Database Modeling
Systems Programming	Computer Systems and Networks	Personal Software Process Principles and practices
Parallel Systems Introduction to parallel systems	Computer Network Security	Individual Software Design and Development
Introduction to Machine Learning	Computing Theory	Introduction to Database Systems

Table 3: Top-5 courses from two CS curricula and one SE curriculum.

Social Implications. Our approach is intended to serve as a tool that supports stakeholders in Education during curriculum design. Completely removing manual inspection and treating our automatic comparisons as ground truths could result in curricula that disregards specific institutional needs. Instead, we recommend stakeholders to use our algorithm together with other specific components (e.g. specialization areas, soft skills, culture, etc). Hence, international standards and regional needs can be combined, supervised by a human expert.

9 Conclusion

In this paper, we explored an intuitive way to generate accurate representations for understanding computing curricula, by combining course-guided attention and metric learning. Our approach finds more cohesive groups with clear separations among them. These groupings are helpful for different machine learning models. We also analyzed what our approach learns via attention weights, topic modeling, and visualization techniques.

10 Limitations

Some limitations of our approach include:

Computational power. Due to using BERT (Devlin et al., 2019), our model requires a forward pass of this deep model and GPU infrastructure for faster prediction.

Inability to associate a core course to a specific computing career. Our model can identify core courses in general, however it cannot identify importance per computing career. For example, "Advanced Algorithms" is more important to a Computer Science curriculum than to Information System. Unfortunately, our model is not able to distinguish it. In future work, we plan to develop an attention module per computing career.

References

Bob Adamson and Paul Morris. 2007. Comparing curricula. In *Comparative Education Research: Approaches and Methods*, pages 263–282, Dordrecht. Springer Netherlands.

Renata Araujo, Héctor Cancela, Beatriz Florian-Gaviria, Ernesto Cuadros-Vargas, and John Impagliazzo. 2020. Toward a common understanding on computing curricula in latin america. In *2020 IEEE World Conference on Engineering Education (EDUNINE)*, pages 1–5.

Ernesto Cuadros-Vargas, Antonio Silva-Sprock, Danet Delgado-Castillo, Yosly Hernández-Bieliukas, and César Collazos. 2013. Evolution of the computing curricula for computer science in latin america 2013. In *2013 XXXIX Latin American Computing Conference (CLEI)*, pages 1–10.

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)*.

Patrick Föll and Frédéric Thiesse. 2021. Exploring Information Systems Curricula. *Business & Information Systems Engineering: The International Journal of WIRTSCHAFTSINFORMATIK*, 63(6):711–732.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. arXiv.

Kornraphop Kawintiranon, Peerapon Vateekul, Atiwong Suchato, and Proadpran Punyabukkana. 2016. Understanding knowledge areas in curriculum through text mining from course materials. In *2016 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, pages 161–168.

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

Yoshitatsu Matsuda, Takayuki Sekiya, and Kazunori Yamaguchi. 2018. Curriculum analysis of computer science departments by simplified, supervised lda. *Journal of Information Processing*, 26:497–508.

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 J. Belongie, and Ser-Nam Lim. 2020. Pytorch metric learning. *CoRR*, abs/2008.09164.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. 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 *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019 (NeurIPS 2019)*, pages 8024–8035.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

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, Barbara Viola, John Impagliazzo, Renzo Angles, Mariela Curiel, Paul Leger, Jorge Murillo, Hernán Nina, José Antonio Pow-Sang, and Ignacio Trejos. 2016. Latin american perspectives to internationalize undergraduate information technology education. In *Proceedings of the 2016 ITiCSE Working Group Reports*, ITiCSE '16, page 1–22, New York, NY, USA. Association for Computing Machinery.

Claude Sammut and Geoffrey I. Webb, editors. 2010. *TF–IDF*, pages 986–987. Springer US, Boston, MA.

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)*, pages 815–823.

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

Shingo Takada, Ernesto Cuadros-Vargas, John Impagliazzo, Steven Gordon, Linda Marshall, Heikki Topi, Gerrit Veer, and Les Waguespack. 2020. Toward the visual understanding of computing curricula. *Education and Information Technologies*, 25.

Jason West. 2017. Validating curriculum development using text mining. *The Curriculum Journal*, 28(3):389–402.

Eric P. Xing, Andrew Y. Ng, Michael I. Jordan, and Stuart Russell. 2002. Distance metric learning, with application to clustering with side-information. In *Proceedings of the 15th International Conference on Neural Information Processing Systems*, NIPS'02, page 521–528, Cambridge, MA, USA. MIT Press.

A Appendix

We provide curriculum samples, 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, comparative word cloud analysis with TF-IDF, and attention weights for the best baseline competitor.

A.1 Curriculum Sample

Each collected curriculum in our dataset consists of a set of courses, and each course has an associated title and description. We depicted an example from each computing program in Table 4.

A.2 Quantitative experiments

A.2.1 Range parameters for experiments

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

- For k-nearest neighbour (KNN), we evaluate k with values [3,5,7].
- For Logistic Regression (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 Qualitative experiments

A.3.1 Topic modeling

As a complementary way to understand our selected courses, we selected the ten course titles with highest attention, and input them to BART topic model (Popa and Rebedea, 2021)[7] to name them.

The named topics are shown in Table 5. 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 curriculums.

A.3.2 Counts for attended courses

Figure 6 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.

[7] 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 4: Sample curriculum showing course titles and their description per computing career.

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

Table 5: Topic identified with each computing curriculm using BART (Popa and Rebedea, 2021) model.

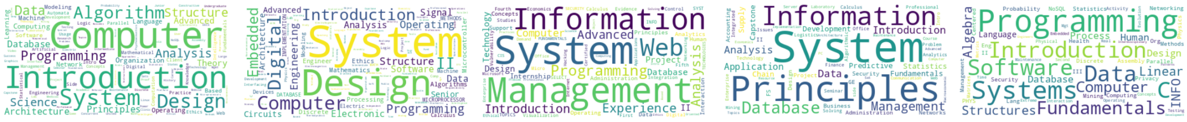
In summary, IT and IS are related to management and information knowledge. CE focuses on hardware concepts such as systems, design, and digital. Finally, CS and SE focus on software development related to programming, data, and algorithm courses.

A.3.3 Word cloud TF-IDF vs Course-guided attention

We compare word clouds from $TF - IDF$ with our approach $Bert_{met+att}$ in Figure 5. We observe that *TF-IDF* word clouds are more pollute with non-related words as opposite to our approach. Also, some important words such as: *programming*, *analysis*, *structure* are less predominant for CS in the TF-IDF word cloud. A extreme case can be seen in SE, where is hard to identify predominant words for TF-IDF. In contrast, our approach identifies important words: *Software*, *programming*, *systems*, *data*, etc. For CE, our approach shows a strong relationship between *digital*, *computer*, *design* among others than *TF-IDF*. For IT and IS, TF-IDF does not show all important words such as: *management*, *information*, and *web*. Finally, for all categories, *TF-IDF* can identify relevant words, but also present meaningless ones such as *course*, *student*, including, among others.

A.3.4 Attention weights best competitor

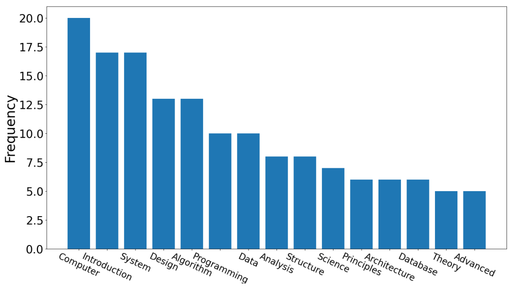
We analyze our best competitor $Fusion_{met+att}$ to discover interesting knowledge. We extract attention weights for each embedding representation. On average, we obtained 0.2149 weight for $Glove$, 0.0621 for $Word2vec$, and 0.7230 for $Bert$. This finding confirms our election to select $Bert$ embedding 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 rows 5 to 7 in Table 2).



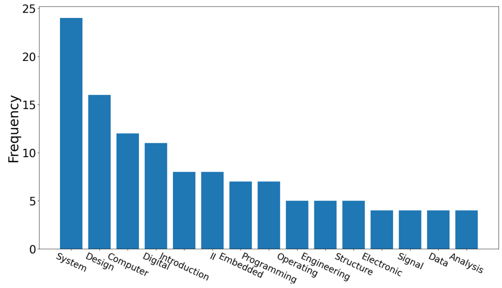
(a) CS$_{Bert_{met+att}}$ (b) CE$_{Bert_{met+att}}$ (c) IT$_{Bert_{met+att}}$ (d) IS$_{Bert_{met+att}}$ (e) SE$_{Bert_{met+att}}$

(f) CS$_{TF-IDF}$ (g) CE$_{TF-IDF}$ (h) IT$_{TF-IDF}$ (i) IS$_{TF-IDF}$ (j) SE$_{TF-IDF}$

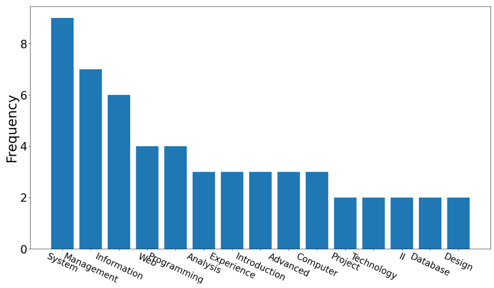
Figure 5: Word cloud comparison of $Bert_{met+att}$ and $TF - IDF$ among five computing careers.



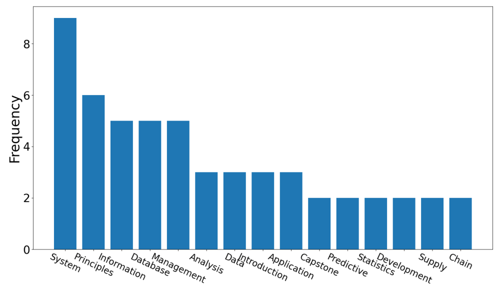
(a) CS



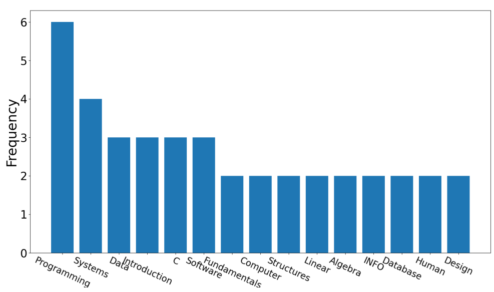
(b) CE



(c) IT



(d) IS



(e) SE

Figure 6: Top fifteen frequency terms of each category. The X-axis shows the word term from course titles, 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).