A SUPPORT TOOL TO IMPROVE COURSE CREDIT TRANSFER IN AN EDUCATION INSTITUTION

Uma Ferramenta de Suporte para Melhorar o Processo de Transferência de Créditos de Curso em uma Instituição Educacional

Una Herramienta de Apoyo para Mejorar el Proceso de Transferencia de Créditos de Cursos en una Institución Educativa

Revista Desafios

Artigo Original
Original Article
Artículo Original

Marcelo Lisboa Rocha[*1], Denis da Silva Passos[2], David Nadler Prata[3], Luís Eduardo Bovolato[4], Diego Paixão Pinheiro[5]

[1]Programa de Pós-Graduação em Modelagem Computacional de Sistemas, Universidade Federal do Tocantins, Palmas, Tocantins, Brasil.

[2]Programa de Pós-Graduação em Modelagem Computacional de Sistemas, Universidade Federal do Tocantins, Palmas, Tocantins, Brasil.

[3]Programa de Pós-Graduação em Modelagem Computacional de Sistemas, Universidade Federal do Tocantins, Palmas, Tocantins, Brasil.

[4]University Dean. Federal University of Tocantins, Palmas, Tocantins, Brasil.

[5]Curso de Ciência da Computação, Universidade Federal do Tocantins, Palmas, Tocantins, Brasil.

**Correspondence: Programa de Pós-Graduação em Modelagem Computacional de Sistemas, Universidade Federal do Tocantins, Av. NS 15, 109 Norte, Palmas, TO, Brasil. CEP:77.010-090. E-mail mlisboa@uft.edu.br.*

Artigo recebido em 08/11/2021 aprovado em 03/05/2022 publicado em 24/05/2022.

ABSTRACT

Processes of course transfer equivalencies should verify the compatibility or equivalence between these curricular components. In educational institutions, the teachers evaluate manually such decision processes with no type of technological support. In order to determine if the courses attended by the students in their institutions of origin can be accepted, the teachers make comparisons between the contents of both courses (attended and requested). Allied to this, the semiannual volume of these processes makes the analysis tedious, time-consuming, error-prone, and constantly challenged by stakeholders. Thus, this work purposes the development of a decision tool based on Natural Language Processing (NLP) techniques to aid in identifying the equivalence of disciplines through the analysis of their contents. The purpose of the decision tool is to support teachers during the evaluation of processes to take advantage of these curricular components. In order to evaluate the performance of the system, we constructed a dataset containing teacher evaluations in real processes of course equivalencies. This dataset was the gold standard (benchmark) for the computational tests. The metrics used in the tests for the evaluation of the proposed technique included AUROC curve, Accuracy and F-Measure.

Keywords: Course equivalence, Natural language processing, Textual similarity.

RESUMO

Os processos de equivalência de transferência de curso devem verificar a compatibilidade ou equivalência entre estes componentes curriculares. Nas instituições de ensino, os professores avaliam manualmente tais processos de decisão sem nenhum tipo de suporte tecnológico. Para determinar se os cursos frequentados pelos alunos nas suas instituições de origem podem ser aceites, os professores fazem comparações entre os conteúdos dos dois cursos (frequentados e solicitados). Aliado a isso, o volume semestral desses processos torna a análise tediosa, demorada, sujeita a erros e constantemente desafiada pelos stakeholders. Assim, este trabalho objetiva o desenvolvimento de uma ferramenta de

decisão baseada em técnicas de Processamento de Linguagem Natural (PNL) para auxiliar na identificação da equivalência de disciplinas por meio da análise de seus conteúdos. O objetivo da ferramenta de decisão é apoiar os professores na avaliação dos processos de aproveitamento destes componentes curriculares. Para avaliar o desempenho do sistema, construímos um conjunto de dados contendo avaliações de professores em processos reais de equivalências de cursos. Este conjunto de dados foi o padrão ouro (benchmark) para os testes computacionais. As métricas utilizadas nos testes de avaliação da técnica proposta incluíram curva AUROC, Exatidão e F-Measure.
Palavras-chave: Equivalência de cursos, Processamento de linguagem natural, Semelhança textual.

RESUMEN
Los procesos de equivalencias de transferencia de cursos deben verificar la compatibilidad o equivalencia entre estos componentes curriculares. En las instituciones educativas, los docentes evalúan manualmente dichos procesos de decisión sin ningún tipo de soporte tecnológico. Para determinar si los cursos a los que asisten los estudiantes en sus instituciones de origen pueden ser aceptados, los docentes realizan comparaciones entre los contenidos de ambos cursos (cursados y solicitados). Aliado a esto, el volumen semestral de estos procesos hace que el análisis sea tedioso, lento, propenso a errores y constantemente desafiado por las partes interesadas. Así, este trabajo tiene como objetivo el desarrollo de una herramienta de decisión basada en técnicas de Procesamiento del Lenguaje Natural (PNL) que ayude a identificar la equivalencia de disciplinas a través del análisis de sus contenidos. El propósito de la herramienta de decisión es apoyar a los docentes durante la evaluación de procesos para aprovechar estos componentes curriculares. Para evaluar el desempeño del sistema, construimos un conjunto de datos que contiene las evaluaciones de los maestros en procesos reales de equivalencias de cursos. Este conjunto de datos fue el estándar de oro (punto de referencia) para las pruebas computacionales. Las métricas utilizadas en las pruebas para la evaluación de la técnica propuesta incluyeron curva AUROC, Precisión y Medida F.
Descriptores: Equivalencia de cursos, procesamiento del lenguaje natural, similitud textual.

INTRODUCTION

Natural Language Processing (NLP) has been utilized in many different ways in the area of education. Among these applications, we can mention: evaluation and monitoring (PRATA et al., 2008), automated essay scoring (BURSTEIN, 2003) and Intelligent Tutoring (POLSON & RICHARDSON, 2013). In those examples, we have a sample of majority application of NLP in core activities. However, NLP techniques can be very useful in building support tools to educational management or support activities.

According to a report released in 2005 by the US National Center for Education Statistics[1] between 1999 and 2000, 59% of egresses of bachelor's degrees of that country attended more than one institution during their academic education (PETER & CATALDI, 2005). In 2010 a study conducted by the National Association for College Admission Counseling reported that one out of three North American university students transferred from one institution to another (FLAGEL, 2010; YANG, 2012). Because of these transfers, an important and very difficult step arises: course credit transferring process.

This issue of courses credit' transfer affects all educational institutions around the world, including Federal University of Tocantins (UFT), a university of North Region of Brazil in legal Amazon. In the UFT, processes of course credit transfer are coming not only from enrollments via transfers. Section III of the university's Academic Regulations provides about university Performance of Curricular Content. Article 90 Sole Paragraph lists the possibilities: "Art. 90 - Sole Paragraph: It will be assured the right of acceptance of curricular content to the student who: I-continues his

[1] Each one of these students brings up in their academic record many attended courses and try to get as many course waivers as possible (acceptances);

studies in the course he/she is enrolled or have re-entered in; II–enter as a graduate; III-have been transferred; IV-have changed the course" (UFT, 2004).

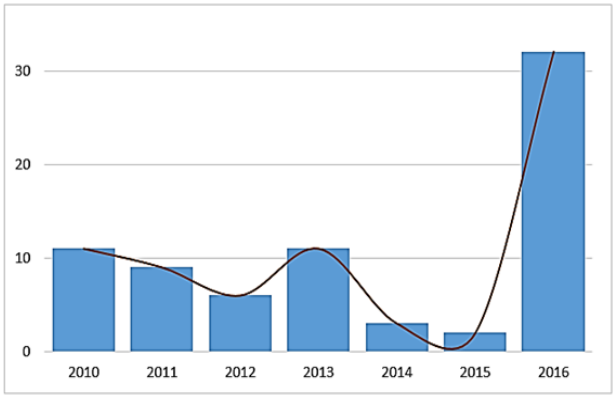
In the UFT undergraduate courses, between 2010 and 2016, more than 70 students entered through one of the ways that typically produce acceptance process: graduate, re-enrollment, course re-option, external transfer and internal transfer (see Fig. 1). Two points worth emphasizing regarding this figure:

1. Each one of these students brings up in their academic record many attended courses and try to get as many course waivers as possible;
2. Besides the enrollment ways presented, a considerable part of the course credits transfer arises from regular selective processes (Examples: entrance examination, SISU – Unified Selection System of Brazilian government to enroll in college institutions), cases in which the student has already attended courses in another higher education institution but entered UFT as a freshman.

Thus, the number of course credit transfer processes is considerably higher than the number of entrants in the presented ways and it does not depend exclusively on it.

In order to determine if the courses already taken by students in their original institution can be accepted, teachers and coordinators must then manually compare the contents of those courses and the contents of pleaded (requested) courses. This process of verifying course equivalency can be exhausting, time-consuming and error-prone. Thus, the main objective of this work is to develop a support tool, utilizing NLP techniques, that is able to automatic identify courses equivalence through the analysis of their contents, to improve the performance of the entire process by providing support to the evaluating teachers.

Figure 1. Graduate registration, re-enrollment and transfers in UFT between 2010 and 2016.



Issue Overview

As explained, UFT Academic Regulations guarantee the acceptance of subjects. However, the following conditions must be met: "Art. 94: The student will be exempted in full when there is a 100% (hundred percent) equivalence of program content and at least 70% (seventy percent) of the workload or 70% (seventy percent) of the program content and 100% (one hundred percent) of the workload" (UFT, 2004).

Thus, given two disciplines for verification, full acceptance (*ai* = *TRUE*) will be granted as follows:

$$ai = \begin{cases} TRUE, IF\ [equiv(chc, chp) \geq 70\%]\ and\ [equiv(ec, ep) = 100\%] \\ TRUE, IF\ [equiv(chc, chp) = 100\%]\ and\ [equiv(ec, ep) \geq 70\%] \\ FALSE, ELSE \end{cases} \quad (1)$$

Where:

- *equiv(x, y)*: Equivalence percentage of contents between course *x* and *y*;
- *ai*: Full acceptance;
- *chc*: Workload of attended course (origin institution);
- *chp*: Workload of requested course (destination institution);

- *ep*: Content of requested course (destination institution).

However, we can empirically state that the evaluation is made considering the following question: *how much of the requested course attributes is contained in the respective attributes of the attended course*?

We therefore have:

$$ai = \begin{cases} TRUE, IF\ (chc \geq (chp \times 0.7))\ and\ (ec = ep) \\ TRUE, IF\ (chc = chp)\ and\ (ec \geq (ep \times 0.7)) \\ FALSE, ELSE \end{cases} \quad (2)$$

As such, we can affirm that making the compatibility computation of the workload is a simple task, given the purely mathematical nature of the operation and of the variables. On the other hand, the courses' contents are short texts usually similar to short paragraphs. Thus, determining a percentage of similarity between the contents of a pair of courses depends exclusively on the perception and experience of the evaluating teacher. Such a peculiarity inevitably gives a certain degree of subjectivity to the process.

Rationale

Currently, the teachers without any type of technological support evaluate the courses credit transfer processes in the scope of UFT manually. Furthermore, the semiannual volume of these processes makes this analysis time consuming, error-prone and constantly challenged by the stakeholders.

From 2010 to 2016, only at Gurupi Campus, 3,405 courses credit transfer were registered in SIE[2].

It's clear that it's desirable the development of a tool utilizing NLP techniques to approach the issue in question, providing automatic support to the evaluation of course credit transfer processes.

PROPOSED APPROACH

This chapter presents the approach that, backed by the literature review and other studies carried out, was utilized to achieve the objectives of this work. The proposal is characterized as hybrid, because it is based on both lexical-morphological similarity and semantic similarity for the calculation of the final equivalence score of the course contents.

Fig. 2 shows the approach, which has its pre-processing phase composed of empirical steps used in Natural Language Processing.

Pre-processing

The main objective of the pre-processing stage is to increase the data initial quality of the texts of both course contents. It is a computationally expensive step because several techniques are usually combined in the form of a pipeline, that is, the output of one is the input to the other (AGGARWAL & ZHAI, 2007).

At the end of this step, these processes will allow a more efficient application of the algorithms selected for both the determination of lexical-morphological similarity and semantic similarity. In this work, the techniques used to perform the pre-processing are shown in following sections.

Conversion to Lower Case

This step simply converts all the text from the course contents to lowercase.

Atomization (Tokenization)

The second step is to convert the text to its minimum lexical units. For this to take place, it is necessary to determine the limits of the lexical units. In Portuguese, the natural delimiter is the blank space. Each lexical unit is called token, reason why this

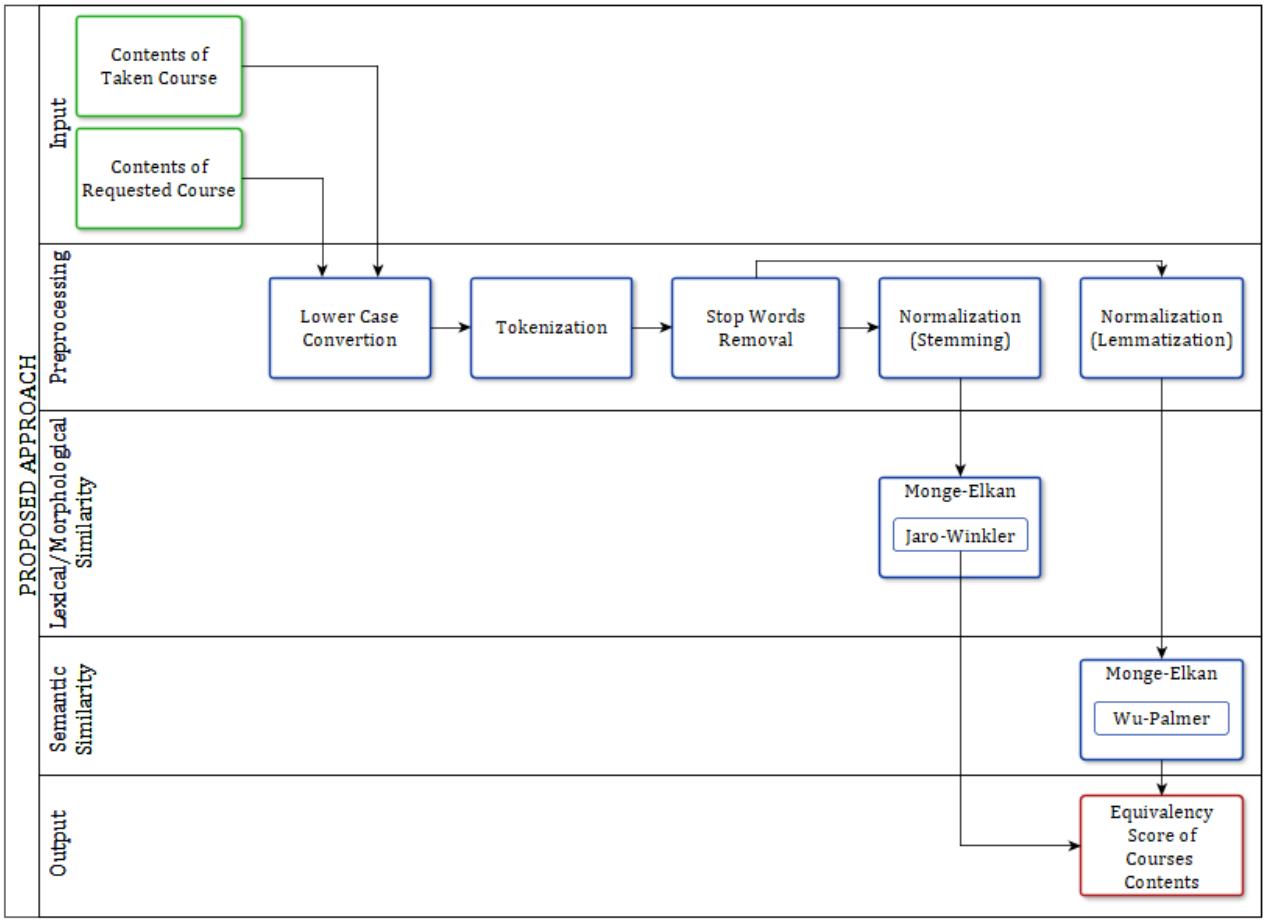
[2] Academic management system utilized by UFT.

process is called tokenization (REHMAN et al., 2013).

Removal of Irrelevant Terms (Stop Words)

In a body of text, several tokens of lesser semantic importance are present, which are called stop words. A stop list is formed by the words of greatest appearance in a textual mass and, usually, corresponds to the articles, conjunctions, prepositions, pronouns and auxiliary verbs of a language (AGGARWAL & ZHAI, 2007).

Figure 2. Overview of proposed approach.



This work utilizes a stop list in Portuguese formed by 203 words appearing in the Natural Language Toolkit (NLTK)[3]. In addition to the stop words, punctuation characters were also removed.

Standardization

In the computational processes of textual analysis, it is common to use different techniques for standardization of linguistic variations. Following will be shown the two standardization techniques used in this work: stemming and lemmatization.

As detailed in Fig. 3, the pre-processing flow takes two distinct ways after the stop words are removed: for lexical-morphological analysis the flow is directed by the sub-stage of radicalization while for the semantic analysis the tokens make their way through the process of lemmatization.

[3] NLTK is a platform for building Python programs to work with human language data. It provides interfaces to over 50 corpora and lexical resources such as WordNet, along with a NLP suite of text processing libraries (BIRD et al., 2009)

Such a division is due to the fact that radicalization empirically brings better results when utilized in conjunction with similarity techniques based on strings (e.g. Jaro-Winkler). On the other hand, in order for the semantic analysis algorithm to function properly, it is necessary for the words to be in their canonical form[4], since the base used (WordNet) thereby stores its synsets[5].

Radicalization. Radicalization (also called stemming) consists of reducing the variations of each word in the text by removing affixes[6] . In this work, radicalization of the tokens has been obtained with the use of the Portuguese Language Suffix Remover (PLSR) - PLSR Stemmer (HUYCK & ORENGO, 2001).

PLSR algorithm is formed by 8 sequential steps according to Fig. 4. Each step has a set of rules (in a total of 253) and only one rule can be applied in each of those steps.

Lemmatization

Another technique related to standardization is the reduction to the canonical form, known as lemmatization. According to BALAKRISHNAN & LLOYD-YEMOH (2014), lemmatization is also defined as the act of representing words in a reduced way.

In the proposed approach, the process of lemmatization is carried out by searching in a dataset. The file used is composed of 850264 pairs of tokens in Portuguese language available by MECHURA (2016).

Determination of Lexical - Morphological Similarity

Lexical or morphological analysis is focused on the study of words, to which (HIPPISLEY, 2010) conferred the status of bricks for the construction of texts in natural language.

In NLP, one of the possible approaches while dealing with lexical units is to treat them as strings (HIPPISLEY, 2010). Thus, we can make use of string-based similarity measures.

In this work, the calculation of Lexical-Morphological Similarity was carried out using the Jaro-Winkler algorithm (WINKLER, 1990) encapsulated in the Monge and Elkan method (MONGE & ELKAN, 1996).

The score of the Jaro-Winkler algorithm (which here we will call *jw*) is based on the number and order of common characters among tokens favoring those with common prefixes. Thus, it is possible to get around small deviations in writing (e.g. basic spelling mistakes). Such behavior seems to be favorable to the proposed application, given that after the radicalization process there may be subtle variations between the radicals of the words of the same lexical family[7]. Usually this happens due to understemming or overstemming[8] of the algorithm utilized.

[4] See Section related to Lematization

[5] Synset instances are the groupings of synonymous words that express the same concept. Some of the words have only one Synset and some have several.

[6] Affixes are morphic elements added to a root or radical in order to modify the meaning of a word. Affixes are subdivided in suffixes (attached to the beginning of the radical) and prefixes (attached to the end of the radical) (BALAKRISHNAN & LLOYD-YEMOH, 2014).

[7] Set of words that share the same radical and are etymologically and morphologically related (BAUER, NATION, 1993).

[8] Errors associated with the radicalization process can be divided into two groups: overstemming (when the removed string is not a suffix but part of the radical) and understemming (when the suffix is not entirely removed) (ORENGO & HUYCK, 2001).

The general score of the Lexical-Morphological analysis is calculated by the Monge-Elkan method using, in this case, the Jaro-Winkler algorithm as an internal similarity function to compare token to token.

Besides the fact of being able to utilize an internal function, the choice of the Monge-Elkan method in this work was given by its asymmetric property.

Figure 3. Proposed pre-processing flow.

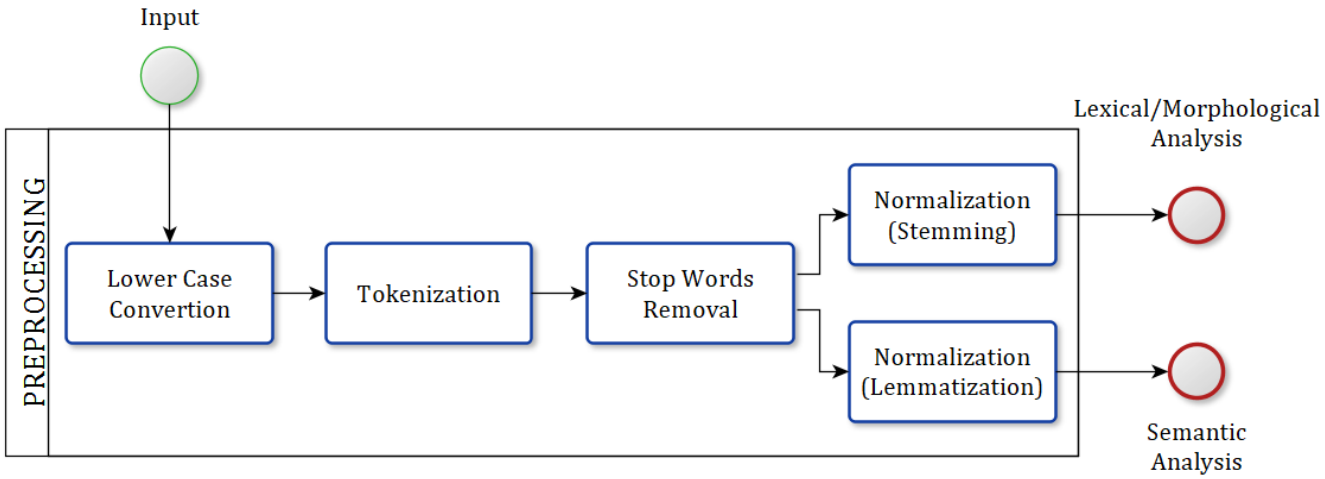
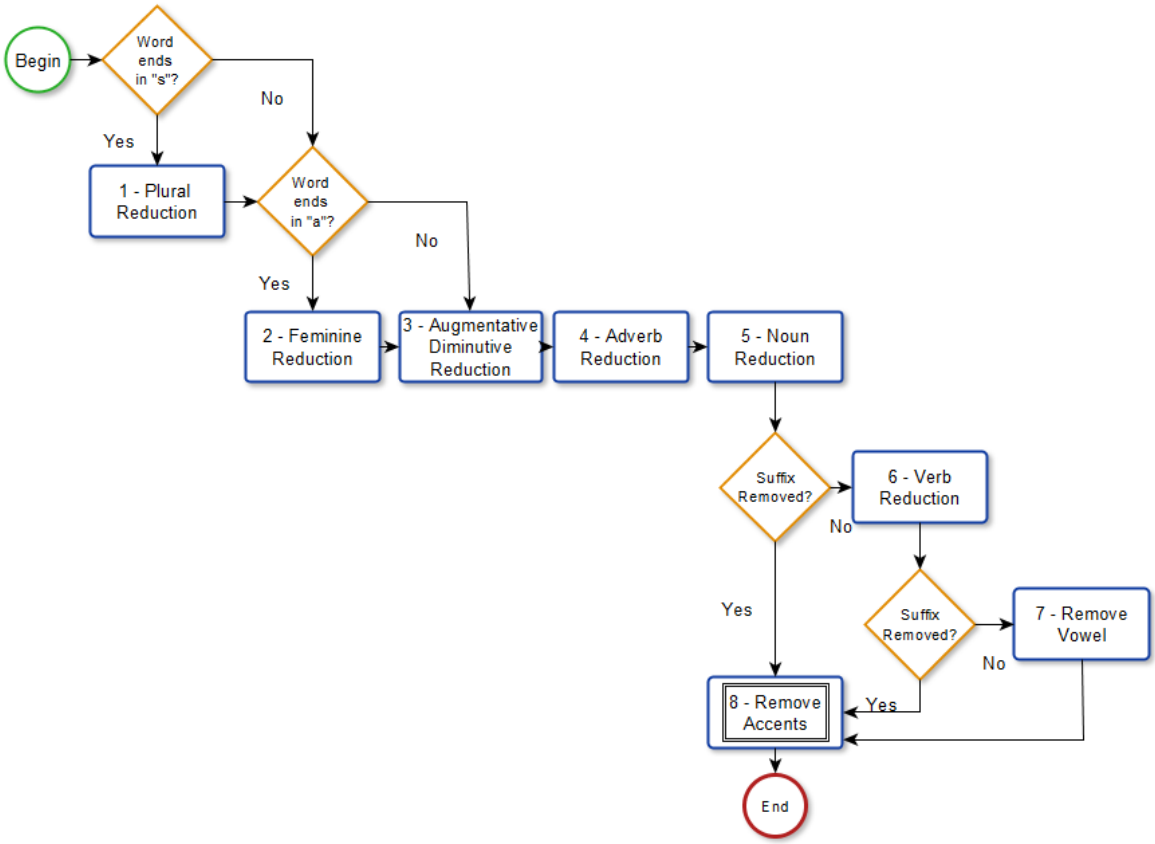


Figure 4. Steps of PLSR Stemmer. Source from (HUYCK & ORENGO, 2001).



Monge-Elkan method asymmetric property is fundamental to the problem addressed by it, when checking if two contents are equivalent, what we want to know is how much of the requested content was

covered by the attended content. That is, how much of the set of tokens of the target content is contained in the set of the original content. This need is characterized as a barrier to the application of most of textual similarity techniques, since they are symmetrical in their majority.

So, we have:

$$damonge\_elkan(DP, DC) \neq monge\_elkan(DC, DP) \quad (3)$$

Where:

1. *DC*: Set of tokens of attended content (Original course);
2. *DP*: Set of tokens of requested content (Destination course).

At this stage, a similarity of 1 (or 100%) would mean a total overlap between sets (DC ⊆ DP) and not that they are identical.

Finally, we can demonstrate the calculation of lexical-morphological similarity as:

$$sim\_lex = \frac{1}{|DP|}\sum_{i=1}^{|DP|}\max\left[jw\left(DP_i, DC_j\right)\right]_{j=1}^{|DC|} \quad (4)$$

For the purpose of a short demonstration, consider the following preprocessed sets of tokens:

- *DP* ← {"endomembr", "biolog"}
- *DC* ← {"biolog", "membr"}.

From DC and DP we have the following elements:

- DP_1 ← "endomembr"
- DP_2 ← "biolog"
- DC_1 ← "biolog"
- DC_2 ← "membr"

So, each element of *DP* (DP_1 to DP_n) is compared to each element of DC (DC_1 to DC_n) by means of internal similarity function (in this case, Jaro-Winkler):

- $jw(DP1, DC1) \leftarrow 0.42592$
- $jw(DP1, DC2) \leftarrow 0.54074$
- $jw(DP2, DC1) \leftarrow 1.00000$
- $jw(DP2, DC2) \leftarrow 0$

In summary, for the application of internal similarity function, the result is the Cartesian product between the sets DP and DP, which results in all possible pairs of elements (as defined in Equation 5):

$$DP \times DC = \{(x, y)|x \in DP, y \in DC\} = \{(DP1, DC1);\ (DP1, DC2);\ (DP2, DC1);\ (DP2, DC2)\} \quad (5)$$

Thus, Monge-Elkan function is presented in Equation 6:

$$monge_{elkan(DP,DC)} = \frac{1}{|DP|}\{max[jw(DP_1, DC_1), jw(DP_1, DC_2)] + max[jw(DP_2, DC_1), jw(DP_2, DC_2)]\} \quad (6)$$

Replacing the values, we have:

$$monge\_elkan(DP, DC) = \frac{1}{2}\{max[0.42592, 0.54074] + max[1.00000, 0]\}$$

So, the final value of *monge_elkan(DP, DC) = 0.77037*. Functioning of Monge-Elkan shown in Equation 4 is shown under algorithmic form in Fig. 5.

Determination of Semantic Similarity

In the area of Linguistics, semantics deals with the meaning of words, phrases, complete sentences and contextualized statements, the latter closer to

pragmatics[9] (REHMAN et al., 2013). From the same point of view, VOORHEES (1999) states that semantic processing is characterized as one of NLP's biggest challenges.

The final goal of a semantic analysis is to understand the formulation: not only reading what is written, but understanding the statement (GODDARD & SCHALLEY, 2010). That way, two phrases with different symbolic and structure information can convey the same meaning or similar meanings. However, if the structures of two sentences are similar, they are more likely to convey similar meanings.

The structural relations mentioned by SRAVANTHI & SRINIVASU (2010) include degrees of relationship and semantic distance between words. Such characteristics are computationally achievable thanks to features as WordNet (FELLBAUM, 1998).

Here, similarly to the calculation of lexical-morphological similarity, the final score calculation of semantic similarity (*sem_sim*) will be carried out using the Monge-Elkan method. However, as an internal function, the Wu and Palmer algorithm (WU & PALMER, 1994) was utilized, as is supported by OpenWordNet-PT (DE PAIVA et al., 2012), as in Equation 7.

$$\text{sem\_sim} = \frac{1}{|DP|}\sum_{i=1}^{|DP|} \max[wuPalmer(DP_i, DC_i)]_j^{|DC|} = 1 \quad (7)$$

The algorithm of the module implemented to the calculation of semantic similarity will be demonstrated in section Main Modules Implemented.

Course Content Equivalence Score

Once the scores of the lexical-morphological similarities (*sim_lex*) and semantics (*sem_sim*), were obtained, it was chosen to calculate the equivalence score[10] between the course content by the maximum value between *sim_lex* and *sem_sim*, as in Equation 8*:*

$$score\_equivalence = max(sim\_lex(Dp, DC), sem\_sim(DP, DC)) \quad (8)$$

PROPOSAL IMPLEMENTATION OF SUPPORT TOOL

Here we aim to describe the methods utilized to implement a tool to integrate the phases of the approach described in section of Proposed Approach. This tool also aimed to facilitate data entry for the computational tests that will be presented in section Experiments and Computational Results to accomplish the general objective of this work.

Definition of Technologies

For the natural language processing, the Python language and the Natural Language Toolkit (NLTK) library (BIRD, KLEIN & LOPER, 2009) were used. NLTK is a platform for building Python programs to work with human language data. The platform provides interfaces to over 50 corpora and lexical resources, such as WordNet, along with a set of text processing libraries for classification, atomization, radicalization, syntactic labeling and semantic analysis among other possibilities (BIRD, KLEIN & LOPER, 2009). In addition to WordNet in English, NLTK brings the equivalents in other languages, such as OpenWordNet-PT (DE PAIVA et

[9] Use of language in different contexts and how they affect its meaning and interpretation (Saint-Dizier, 1998).

[10] It should be noted that for correct use of *score_equivalence* in reports, an adjustment factor (fa) should be applied over the resulting value. This factor will be shown in section related to Adjustment Factor for the Course Content Equivalence Score.

al., 2012) which was utilized in this work. NLTK is licensed by the Apache License Version 2.0 and is freely used for development and commercialization.

Main Modules Implemented

Here, the two main modules implemented to achieve the work objectives are described.

Lexical-Morphological Analyzer

As stated before, the determination of lexical-morphological similarity will be carried out by applying the Monge and Elkan method (MONGE & ELKAN, 1996) using the Jaro-Winkler algorithm (WINKLER, 1990) as an internal function.

The operation of the lexical-morphological analyzer employing both techniques mentioned above is shown in Figs. 5 and 6, respectively.

The parameters (*course_request* and *course_taken*) of the function shown in Fig. 5, represent the token sets of the content already treated by the three initial steps of pre-processing. The final step, that is, the standardization is carried out according to the type of analysis being run: For the lexical-morphological analysis, the radicalization is carried out through the RSLP Stemmer; for the semantic analysis, the lemmatization is carried out.

The type of analysis to be carried out by the function is defined by the parameter *typer_of_analysis*, which supports two alternatives (lexical-morphological analysis or semantic analysis) as explained in Table 1.

Lines 12 to 23 of the algorithm shown in Fig. 5 reproduce specifically Equation 4 (or Equation 7, depending on the selected option). In lines 12 to 14, the algorithm calculates the Cartesian product between the two sets of tokens *course_requested* and *course_taken*. The Cartesian product will result in all possible combinations between elements of both sets. Thus, through the internal function, the similarity values are obtained for each pair generated by the Cartesian product. In other words, each element in *course_requested* is compared to each element in *course_taken*. The highest similarity value obtained for each element of the *course_requested* is accumulated on variable *sum_of_maximums* (line 20).

Once the value of *sum_of_maximums* is obtained, the Monge-Elkan score is given by the quotient between that value and the size of set of tokens of the requested course content *course_requested*, as in line 23 of Fig. 5.

The algorithm in Fig. 6 shows the functioning of the Jaro-Winkler measure, used as an internal similarity function when the lexical-morphological analysis is carried out.

In addition to the two tokens to be compared *(token_a and token_b)*, the function also receives the weight assigned to the common prefix between the two tokens (*pp*). The default value for the *pp* parameter is 0.1 (WINKLER, 1990).

The measure proposed by WINKLER (1990) is an extension of that developed by JARO (1989). Thus, the algorithm of Fig. 6 previously calculate the Jaro *score*. For the calculation of the Jaro score (line 8) some preliminary operations are carried out such as the count of characters in common between the tokens (line 5) and the count of transpositions (line 6).

When Jaro score is defined, the size of the longest common prefix among tokens (line 10) is verified and the extension proposed by WINKLER (1990) (line 12) is implemented.

Semantic Analyzer

In section related to Lexical-Morphological Analyzer we highlight that the Monge-Elkan method

was utilized for both lexical-morphological analysis and semantic analysis. Thus, a generic function was implemented to carry out both analyzers (already shown in Fig. 5). For that reason, this section will only present the algorithm referring to the internal similarity function used in the semantic analysis, that is, the Wu and Palmer (WU & PALMER, 1994) measure, shown in Fig. 7.

The algorithm consists of the *wuPalmerScore* and *wuPalmer* functions. The first makes use of the second and is responsible for returning the value of the semantic analysis to the generic function on Fig. 5.

Function *wuPalmerScore* (line 1) receives two lemmas from Fig. 5 and collects from OpenWordNet-PT all the concepts available to each. In this way, similarly to the Monge-Elkan method, the Cartesian product is calculated between the pairs of all the concepts obtained. The Wu-Palmer semantic relation between all the pairs of the Cartesian product is then measured by the function *wuPalmer* (line 19) and the highest value obtained to be returned is selected.

The proposed measure in WU & PALMER (1994) is implemented specifically in the *wuPalmer* function that starts at the line 19. Initially the LCA *(Lowest Common* Ancestor) is calculated of the pair of *synsets* received as parameters.

Afterwards, the depth of the LCA in the OpenWordNet-PT hierarchy is calculated, the depths of each *synset* passing through the LCA. Finally, in line 27, the equation proposed by WU & PALMER (1994):

$$wup(C_1, C_2) = 2 * \frac{depth\left(LC\ A\ (C_1, C_2)\right)}{depth\ (C_1) + depth\ (C_2)} \quad (9)$$

Table 1. Selection of analysis to be carried out by the Monge-Elkan function.

Value of type_of_analysis	Type of Analysis	Internal Function
"L" or "l"	Lexical-morphological	jaroWinklerScore()
"S" or "s"	Semantic	wuPalmerScore()

Figure 5. Monge-Elkan method for calculating the lexical-morphological and semantic scores (depending on the selected internal function).

```
   Input:
   course_requested, course_taken, type_of_analysis
 1 begin
 2   switch type_of_analysis do
 3     case "L" or "l" do
 4       internal_function ← jaroWinklerScore()
 5     end
 6     case "S" or "s" do
 7       internal_function ← wuPalmerScore()
 8     end
 9   end
10   sum_of_maximums ← 0
11
12   foreach i ∈ course_requested do
13     i_max_sim ← 0
14     foreach j ∈ course_taken do
15       sim ← internal_function(i, j)
16       if sim > i_max_sim then
17         i_max_sim ← sim
18       end
19     end
20     sum_of_maximums ← sum_of_maximums + i_max_sim
21   end
22
23   score ← sum_of_maximums /
       (number_of_elements(course_requested))
24
25   return score
26 end
```

Figure 6. Jaro-Winkler algorithm (used as internal function in the Monge-Elkan method for performing lexical-morphological analysis).

```
function jaroWinklerScore(token_a, token_b, pw):
    len_token_a ← length(token_a)
    len_token_b ← length(token_b)

    cc ← commonCharacterCounter(token_a, token_b)
    t ← transpositionCounter(token_a, token_b)

    jaro_score ←
     (1/3)*((cc/len_token_a)+(cc/len_token_b)+((cc-(t/2))/cc))

    len_lcp ← length(longestCommonPrefix(token_a, token_b))

    score ← (1 - len_lcp * pw) * jaro_score + len_lcp * pw

    return score
end of function
```

Figure 7. Functions for applying Wu-Palmer algorithm (used in the semantic analysis by Monge-Elkan method).

```
function wuPalmerScore(token_a, token_b):
    all_synsets_a ← getAllConcepts(token_a)

    all_synsets_b ← getAllConcepts(token_b)

    max_score ← 0

    foreach i ∈ all_synsets_a do
        foreach j ∈ all_synsets_b do
            score ← wu_palmer(i, j)
            sim > max_score  max_score ← sim
        end
    end
    return max_score
end of function

function wuPalmer(synset_a, synset_b):
    lca ← getLCA(synset_a, synset_b)

    depth_lca ← getDistance(lca, root)

    depth_a ← depth_lca + getDistance(lca, synset_a)
    depth_b ← depth_lca + getDistance(lca, synset_b)

    score ← (2 * depth_lca)/(depth_a + depth_b)

    return score
end of function
```

EXPERIMENTAL AND COMPUTATIONAL RESULTS

In this section, the results obtained from the computational tests carried out with the purpose of analyzing the performance and validating the operation of the proposed approach as a support tool to improve credit course transfer are presented.

The computational tests were carried out on a computer with Microsoft® Windows ® 10 Home 64-bits operating system, Intel® Core™ i3-4005U CPU @ 1.70 GHz processor and 4096 MB of RAM memory PC3-12800 (800 MHz) DDR3.

Construction of the Experimental Dataset

In order to perform the computational tests, a dataset was created. The data used were collected from the physical archive of the UFT Academic Secretary - Gurupi Campus, based on 40 different course credit transfer processes coming from 31 different Educational Institutions (EIs). This dataset is composed of 100 pairs of course contents with the teacher's opinion about the compatibility between them *opinion_proc*. In addition, the analysis values calculated by the system for each pair of contents were later added to the dataset *score_sys*. This dataset is available at https://tinyurl.com/ydutu8eu.

Table 2 displays a data dictionary for the constructed dataset. The contents of the requested courses were extracted from the pedagogical projects of the destination courses. The contents of the courses attended by the students in their institution of origin were digitized. The *opinion_proc* field corresponds to the evaluation made by the teachers and will serve as a reference value *gold standard.*

For contextualization purposes, with regard to binary data *opinion_proc*, the following convention was adopted:

$$Granted \longleftrightarrow Positive \longleftrightarrow 1$$
$$Refused \longleftrightarrow Negative \longleftrightarrow 0$$

Table 2. Dataset data dictionary used in the tests.

Field	Type	Description
content_course_attended	Text	Content of the course attended by the student in the original institution
content_course_requested	Text	Content of the course in which the student requests waiver on the destination institution
opinion_proc	Binary	Opinion given by the teacher responsible for evaluation of the acceptance. Two values are allowed: Granted (1) or Refused (0)
score_sys	Numeric	Equivalence score between the course contents scores calculated by the system. Continuous variable in a range from 0 to 1.

Valuation Metrics Utilized

The matching between two sets of strings can be seen as a classification problem on the Cartesian product of sets (JIMENEZ et al., 2009). In this work, for performance evaluation purposes, the system will be considered as a binary classifier because only two outputs are possible: *Granted* (valid match) or *Refused* (non-valid match).

For the system to classify a pair of course contents as *Granted* or *Refused*, it is necessary to establish a *(threshold)* λ. Pairs with course contents Equivalence Score *(score_sys)* greater than or equal to *lambda* are labeled as *Granted* (1) and the others as

Refused (0). The variables *opinion_proc*[11], *score_sys*[12] and threshold λ, gives to the system 4 possibilities of results:

- **True Positive - TP:** If it is classified by the system as true (granted) and is in fact true;
- **False Positive - FP:** Classified by the system as granted but refused in the actual process;
- **True Negative - TN:** Classified by the system as refused and in fact denied in the actual process;
- **False Negative - FN:** Classified by the system as refused but evaluated as granted in the actual process.

From this, it is possible to assemble a confusion matrix (Table 3), which is composed of the quantitative of each of the cases already listed. In reality, several confusion matrices may be generated, one for each threshold which is chosen according to the purpose of the experiment.

Once counted in the confusion matrix (see Table 4), results serve as a basis for measures calculation that will provide an overview of the system's effectiveness. In this work, the measures described in section related to Valuation Metrics Utilized, we used them to evaluate the performance of the proposed technique for course content acceptance in course credit transfer process.

Table 3. Confusion matrix for system evaluation.

	opinion_proc	
	granted	**refused**
score_sys ≥ λ	TP	FP
socre_sys < λ	FN	TN

Table 4. Confusion matrix for system evaluation.

	opinion_proc	
λ = 0,79694	**granted**	**refused**
score_sys ≥ λ	TP = 47	FP = 14
socre_sys < λ	FN = 19	TN = 20

ROC Curve and Area under the ROC Curve

One way to provide a broad view of the efficiency of a binary classifier is the Receiver Operating Characteristic (ROC) curves (ZWEIG & CAMPBELL, 1993). ROC technique was developed in the signal processing area and the term 'receiver operating characteristic' refers to the performance (the 'operating characteristic') of the observer (the 'receiver') that assign cases into dichotomous classes.

A ROC curve is created out of confusion matrices based on different λ thresholds. Once the confusion matrix is created for each λ, the false positive rate (*FPR*) and true positive rate *(TPR)* [13]are calculated, according to the following formulas:

$$TPR = \frac{TP}{TP + FN} \quad (10)$$

$$FPR = \frac{FP}{FP + TN} \quad (11)$$

After that, the ROC curve is obtained by the distribution of the values of *FPR* in the axis of the abscissa *(x)* and by the distribution of the values of *TPR* in the axis of the ordinate *(y)*.

Therefore the closer the ROC curve is to the upper left corner, the higher the overall accuracy of the test (ZWEIG & CAMPBELL, 1993).

In the ROC context, the area under ROC Curve (*AUC or AUROC*) ϵ [0, 1] and provides us an insight into the power of discrimination of the model. The greater the area under the ROC curve, the better the classification capacity of the model (HOSMER,

[11] Opinion on the actual process (gold standard). See Table 2.

[12] Course Content Equivalence Score, calculated by the system. See Table 2.

[13] True Positive Rate (TPR) is also called recall or sensitivity.

2000). Table 5 list the values for the AUROC interpretation.

Accuracy

Accuracy ϵ [0, 1][14] is the number of all correct classifications divided by the total number of classifications. The closer to 1.0, the better is the accuracy.

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \quad (12)$$

Accuracy indicates, in broad terms, how frequently the classifier is correct.

F-Measure

As specified in (SOKOLOVA et al., 2006), this measure ϵ [0, 1] and is the harmonic mean between precision and recall (both are explained respectively below). By combining precision and recall, F-measure can be used to evaluate the overall performance of the support tool (DURIC & GASEVIC, 2013). The closer to 1.0 the F-measure, the better is the performance of the classification technique.

$$F - Measure = 2 \times \frac{Precision \times Recall}{Precision \times Recall} \quad (13)$$

Precision. This measure ϵ [0, 1] is the proportion of cases correctly labeled as *Granted$* among the total number of those classified as such (including false positives) (SOKOLOVA et al., 2006). If there are no false positives, the accuracy value is 1.0. This metrics shows how often the system is correctly classifying as *Granted.*

$$Precision = \frac{TP}{TP + FP} \quad (14)$$

Recall. This measure ϵ [0, 1] is the proportion of cases correctly labeled as *Granted* among the total number of positives (whether true or false) (SOKOLOVA et al., 2006). In the absence of false negatives, the value of the recall is 1.0. Depending on the context, recall is also known as a *sensitivity* or *true positive rate - TPR.*

$$Recall = \frac{TP}{TP + FN} \quad (15)$$

This metrics shows how frequently the system labels as *Granted* the cases that in fact are *Granted* in the *gold standard.*

Results achieved

Here, results achieved with the application of the metrics presented in section Valuation Metrics Utilized are presented.

To calculate the area under the ROC curve, the MedCalc®[15] software was used. When calculating AUROC, the software generated a list with the 36 main thresholds λ, and that list was used as the basis for the determination of the other metrics.

For each threshold contained in the list generated by MedCalc®, a corresponding confusion matrix was created (that is, totals of true positives, false positives, true negatives and false negatives were calculated and counted). In total, 36 confusion matrices were generated. For illustration purposes, Table 6 shows part of the amounts accounted for.

Once the confusion matrices were created, it was possible to plot the model's ROC curve, as well as to calculate the accuracy and F-measure values for

[14] $x \epsilon [a, b] \rightarrow \{x \epsilon \mathbb{R} : a \leq x \leq b\}$

[15] MedCalc Statistical Software version 18.2.1. Available at: https://www.medcal.org.

each of the thresholds.

Results: ROC curve and AUROC

In Fig. 8 we have the ROC curve obtained with the system outputs. In Fig. 8, the hatched area corresponds to AUROC with value of 0.7112, which classifies the discrimination of the system as acceptable, according to Table 5.

When analyzing AUROC values, the null hypothesis (H_0) is the area under the curve that has a value equal to 0.5, that is, that the ROC curve is arranged in a diagonal position on the graph, according to the dotted line in Fig. 8 (the closer the curve comes to the 45-degree diagonal, the less accurate the test). In other words, the null hypothesis states that the discrimination capacity of the model is non-existent.

Figure 8. ROC Curve and AUROC.

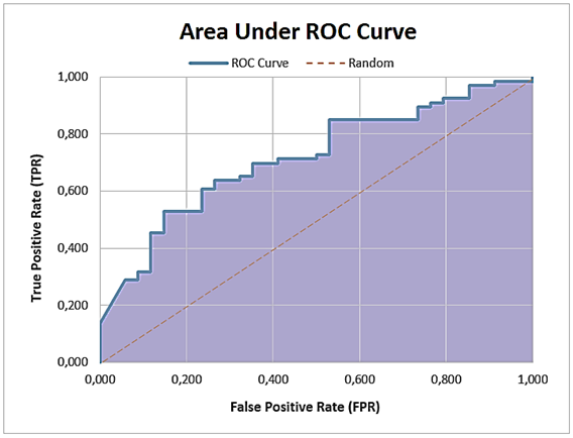


Table 5. Parameters for AUROC interpretation.

	Description
AUROC = 0.5	No discrimination (random process)
0.7 ≤ AUROC < 0.8	Acceptable discrimination
0.8 ≤ AUROC < 0.9	Excellent discrimination
AUROC ≥ 0.9	Outstanding discrimination

Source of HOSMER (2000)

[16] "Loss-and-win" or cost-benefit ratio.

For the proposed model, the obtained *p-value* was **0.0000791**. Therefore, it can be concluded that the area under the ROC curve is significantly different from 0.5. Therefore, refuting the null hypothesis, there is statistical evidence that the system has the ability to distinguish between the two classes (*Granted* and *Refused* for course credit transfer' process).

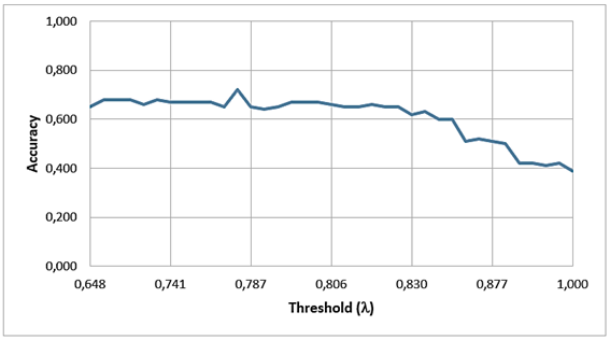
Table 6. Confusion matrices by threshold.

Threshold (λ)	TP	FP	TN	FN
0.64795	64	34	0	1
⋮	⋮	⋮	⋮	⋮
0.77310	56	18	16	10
0.78729	49	18	16	17
0.78925	48	18	16	18
0.79126	48	17	17	18
0.79694	47	14	20	19
0.79697	47	14	20	19
0.80271	46	13	21	20
0.80559	44	12	22	22
⋮	⋮	⋮	⋮	⋮
0.97176	8	0	34	58

Results Related to Accuracy

The system accuracy indices can be observed in Fig. 9. The best indices reached were **0.72** (for λ = 0.77310) and **0.68** (for the thresholds 0.70386, 0.70390, 0.72021, 0.73382).

Figure 9. ROC Curve and AUROC.



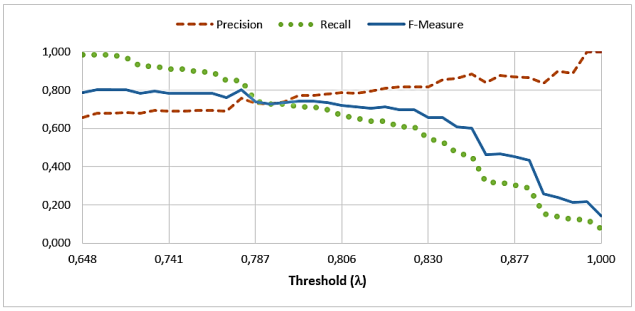
Results Related to F-Measure

Fig. 10 shows *trade-off* [16]between precision

and recall: recall has a decreasing trend, while accuracy as a whole increases. The maximum value reached by F-measure is known as *F1 Score* and is obtained at λ threshold with the best balance between precision and recall (JIMENEZ et al., 2009).

In the proposed system, *F1 Score* value was **0.80247** with λ = 0.70390 or λ = **0.70386**. After *F1 Score*, the highest value reached by F-Measure was **0.8** (for thresholds 0.72021 and 0.77310).

Figure 10. Precision, recall and F-measure obtained by threshold.



Results Analysis

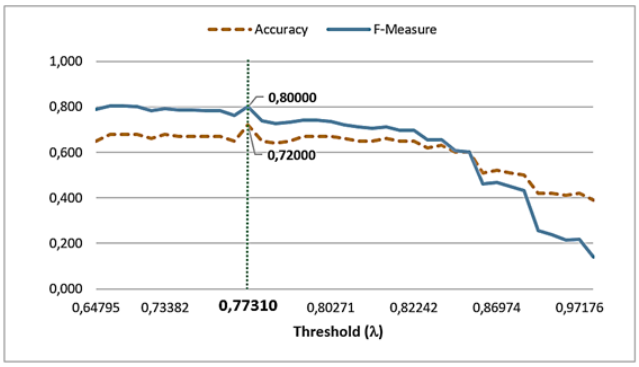
Table 7 lists the λ thresholds that reached the best results for F-Measure and Accuracy. Analyzing data from Table 7 and Fig. 11 it is possible to observe that the best overall results are reached with λ = 0.77310. The accuracy at this threshold is the biggest (0.72000), in addition, the difference between F-Measure of this threshold and the best F- Measure achieved among all thresholds is only 0.00247, which can be noted only from the third decimal place on. We can also highlight that this threshold has the best balance between the totals of *TP*, *FP*, *TN* and *FN*.

As shown in Fig. 12, the analysis of the ROC[17] curve carried out using the MedCalc® software corroborates that the results obtained with λ = 0.77310 are the ones which are closer to human evaluations.

Table 7. Best thresholds for F-Measure and Accuracy.

Threshold (λ)	TP	FP	TN	FN	F-Measure	Acuracy
0.70386	65	31	3	1	0.80247	0.68000
0.70390	65	31	3	1	0.80247	0.68000
0.72021	64	30	4	2	0.80000	0.68000
0.73382	61	27	7	5	0.79221	0.68000
0.77310	**56**	**18**	**16**	**10**	**0.80000**	**0.72000**

Figure 11. Accuracy and F-measure by threshold (highlight in peaks obtained in λ= 0.77310).



Adjustment Factor for the Course Content Equivalence Score

As described in the UFT scope, the minimum coverage percentage for which the content of a course subject is considered equivalent is 70%. That value can be converted to the same scale as the λ thresholds utilized in this work, resulting in 0.70000.

When issuing the opinions contained in the experimental dataset, teachers considered the mark of 70% (or 0.70) as the minimum accepted percentage (*pcUFT*). Thus, once the λ threshold is met with results that are closer to human evaluations λ, it is possible to establish an adjustment factor (*fa*) for the system output:

$$fa = \frac{pcUFT}{\Lambda} = \frac{0.7000}{0.77310} \cong 0{,}90545 \qquad (16)$$

[17] In this analysis, cost 1 was used as a penalty to the False Positive and False Negatives returned by thresholds. The "disease prevalence" parameter, in this case, refers to the percentage of processes contained in the experimental dataset that was evaluated as granted by teachers.

Thus, *Course Content Equivalency Score* (CCES) value can be adjusted before use it to express the system output:

$$CCES = score\_sys \times fa \quad (17)$$

Let us take as an example a pair of course content (attended and requested) which had *score_sys* = 0.56987. By applying the adjustment factor, we should have:

$$CCES = 0.56987 \times 0.90545 \cong 0.51599$$

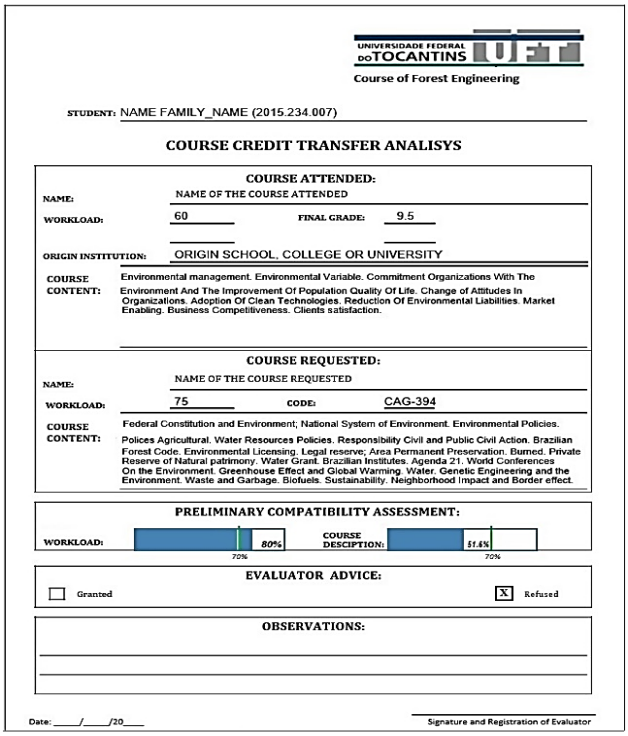
Figure 12. ROC Curve analysis: Optimal criterion.

Optimal Criterion	
Optimal criterion*	0,77310
Sensitivity	84,85
Specificity	47,06
* *Taking into account prevalence (66,0%) and estimated costs:* *cost False Positive: 1; cost False Negative: 1* *cost True Positive: 0; cost True Negative: 0*	

In this case, the support tool output is refuse the course equivalency, in other words, the percentage of coverage of the course content attended over the pleaded course content (or requested) would be approximately 51.6%. Thus, according to Equation 2, that course content equivalency will not be accepted (refused).

Fig. 13 shows the cover model that accompanies course contents to be analyzed by the teachers in the UFT course credit transfer acceptance processes of each discipline. The model shown in this figure represents the system output, presenting one more item than the one currently in use: The field "*Course Content"*[18] in the "*Preliminary Compatibility Assessment"* section of the cover. This prior analysis of workload and course content, through this support tool, will help teachers to evaluate course credit transfer acceptance processes, reducing the time to carry it out and error occurrences.

Figure 13. Cover utilized in course credit transfer processes at UFT.


UNIVERSIDADE FEDERAL DO TOCANTINS

Course of Forest Engineering

STUDENT: NAME FAMILY_NAME (2015.234.007)

COURSE CREDIT TRANSFER ANALISYS

COURSE ATTENDED:

NAME: NAME OF THE COURSE ATTENDED

WORKLOAD: 60 FINAL GRADE: 9.5

ORIGIN INSTITUTION: ORIGIN SCHOOL, COLLEGE OR UNIVERSITY

COURSE CONTENT: Environmental management. Environmental Variable. Commitment Organizations With The Environment And The Improvement Of Population Quality Of Life. Change of Attitudes In Organizations. Adoption Of Clean Technologies. Reduction Of Environmental Liabilities. Market Enabling. Business Competitiveness. Clients satisfaction.

COURSE REQUESTED:

NAME: NAME OF THE COURSE REQUESTED

WORKLOAD: 75 CODE: CAG-394

COURSE CONTENT: Federal Constitution and Environment; National System of Environment. Environmental Policies. Polices Agricultural. Water Resources Policies. Responsibility Civil and Public Civil Action. Brazilian Forest Code. Environmental Licensing. Legal reserve; Area Permanent Preservation. Burned. Private Reserve of Natural patrimony. Water Grant. Brazilian Institutes. Agenda 21. World Conferences On the Environment. Greenhouse Effect and Global Warming. Water. Genetic Engineering and the Environment. Waste and Garbage. Biofuels. Sustainability. Neighborhood Impact and Border effect.

PRELIMINARY COMPATIBILITY ASSESSMENT:

WORKLOAD: 80% 70% COURSE DESCIPTION: 51.4% 70%

EVALUATOR ADVICE:

☐ Granted ☒ Refused

OBSERVATIONS:

Date: ____/____/20____ Signature and Registration of Evaluator

CONCLUSION AND FUTURE WORKS

The present work proposed the development of a support tool based on NLP techniques to aid in the process of identifying the equivalence of courses for course credit transfer through the analysis of their course contents. The purpose of the tool is to offer support to teachers in the evaluation of acceptance processes of this important curricular component, being no longer a tedious, laborious and time-consuming task subject to the interpretation of the evaluator.

The research was carried out on textual similarity techniques aiming to acquire the theoretical basis for the proposal development and conduction of

[18] For demonstration purpose, it was used the same percentage of the example (51.60%).

the evaluation and validation experiments. Next, the Python language and the NLTK library were used to implement the proposed approach.

After the implementation of the proposal, the experimental stage to validate the developed work was initiated. In order to evaluate the performance of the system, was created a dataset consisting of 100 pairs of course contents with the teacher's opinion on equivalence (granted or refused opinion in the course credit transfer acceptance process). Data used to create the dataset were collected from actual credit course transfer acceptance processes registered between the years 2014 and 2017, all filed in the physical archive of UFT Academic Secretary - Gurupi Campus. Then, values from the analysis calculated by the system for each pair of contents were added to the dataset.

Once the experimental dataset was created, it was feasible to calculate the metrics defined to evaluate the system efficiency. A value of 0.7112 was obtained for the area under the ROC curve with *p-value* = 0.0000791, proving statistically that the proposed model has acceptable discrimination power. Accuracy indexes of 0.72 were reached. Besides that, the system achieved values of 0.80247 for F-Measure.

Based on the results obtained, it is possible to use natural language processing (NLP) techniques as support in the identification of the equivalence of disciplines, reducing the teachers' load and increasing the performance of the whole process. Despite the case of this tool was implemented to support Portuguese, this can be easily extended to other languages.

Future Works

As future works, it is expected to improve the graphical web interface usability making in order to make it more user-friendly and intuitive. It is also aimed to develop a mechanism that allows generating system reports (as shown in Fig. 13), such reports will serve as support for teachers in the task of evaluating course contents equivalences. It is also an objective to investigate the variation in the quantitative of true positives, false positives, true negatives and false negatives with regard to different thresholds. In addition, it is intended to test the system with measures of semantic similarity based on informational content. Finally, it is planned to carry out performance tests against the best techniques of literature in standard data used by other authors.

All authors declare that there is no potential conflict of interest regarding this article.

REFERENCES

AGGARWAL, C. C.; ZHAI, C. (Eds.). **Mining text data**. Springer Science & Business Media, 2012.

BALAKRISHNAN, V., & LLOYD-YEMOH, E. **Lecture Notes on Software Engineering 2** (3), p. 262–267, 2014.

BIRD, S.; KLEIN, E.; LOPER, E. **Natural language processing with Python: analyzing text with the natural language toolkit**. " O'Reilly Media, Inc.", 2009.

BURSTEIN, J. The e-Rater® Scoring Engine: Automated Essay Scoring with Natural Language Processing., in: M. Shermis, J. Burstein (Eds.), Automated Essay Scoring: A Cross-Disciplinary Perspective, **Lawrence Erlbaum Associates**, Mahwah, New Jersey, Ch. 7, pp. 113–121, 2003.

DE PAIVA, V.; RADEMAKER, A.; DE MELO, G. Openwordnet-pt: An open brazilian wordnet for reasoning. In: **Proceedings of COLING 2012: Demonstration Papers**. p. 353-360, 2012.

Đurić, Z., & Gašević, D. A source code similarity system for plagiarism detection. **The Computer Journal**, v. 56, n. 1, p. 70-86, 2013.

FELLBAUM, C. WordNet: An electronic lexical database and some of its applications. Cambridge, MA: **MIT Press**, 1998.

FLAGEL, A. Special report on the transfer admissions process. **Arlington, VA: National Associate for College Admissions Counseling. Retrieved October**, v. 10, p. 2011, 2010.

GODDARD, C.; SCHALLEY, A. C. Semantic Analysis, in: INDURKHYA, N.; DAMERAU, F. J. (Eds.), Handbook of Natural Language Processing, 2nd Edition, **Chapman & Hall/CRC**, Boca Raton, Ch. 5, pp. 93–120, 2010.

HIPPISLEY, A. Lexical Aanalysis, in: N. Indurkhya, F. J. Damerau (Eds.), **Handbook of Natural Language Processing**, 2nd Edition, Chapman & Hall/CRC, Boca Raton, Ch. 3, pp. 31–58, 2010.

HOSMER, D. W. Assessing the fit of the model. Hosmer DW; Lemeshow S, eds. **Applied logistic regression**, John Wiley & Sons, p. 160-163, 2000.

HUYCK, C.; ORENGO, V. A. Stemming Algorithm for the Portuguese Language, in: **Proceedings of the 8th International Symposium on String Processing and Information Retrieval -SPIRE 2001**, IEEE Computer Society Press, pp. 186–193, 2001. doi:10.1109/SPIRE.2001.10024.14

JARO, M. A. Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida. **Journal of the American Statistical Association**, v. 84, n. 406, p. 414-420, 1989.

JIMENEZ, S., BECERRA, C., GELBUKH, A., & GONZALEZ, F. Generalized mongue-elkan method for approximate text string comparison. In: **International conference on intelligent text processing and computational linguistics**. Springer, Berlin, Heidelberg. p. 559-570, 2009.

MECHURA, M. Machine-Readable Lists of Lemma-Token Pairs in 23 Languages, 2017. Available at: **https://github.com/michmech/lemmatization-lists**

MONGE, A. E.; ELKAN, C. The Field Matching Problem: Algorithms and Applications, in: **Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD'96, AAAI Press**, pp. 267–270**,** 1996.

PETER, K.; CATALDI, E. F. The Road Less Traveled? Students Who Enroll in Multiple Institutions. Postsecondary Education Descriptive Analysis Report. NCES 2005-157. **National Center for Education Statistics**, 2005.

POLSON, M. C.; RICHARDSON, J. J. **Foundations of intelligent tutoring systems**. Psychology Press, 2013.

PRATA, D.; LETOUZE, P.; COSTA, E.; PRATA, M.; BRITO, G. Dialogue analysis in collaborative learning. **International Journal of e-Education, e-Business, e-Management and e-Learning**, v. 2, n. 5, p. 365, 2012.

REHMAN, Z.; ANWAR, W.; BAJWA, U. I.; XUAN, W.; CHAOYING, Z. Morpheme matching based text tokenization for a scarce resourced language. **PloS one**, v. 8, n. 8, p. e68178, 2013.

SOKOLOVA, M.; JAPKOWICZ, N.; SZPAKOWICZ, S. Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In: **Australasian joint conference on artificial intelligence**. Springer, Berlin, Heidelberg. p. 1015-1021, 2006.

SRAVANTHI, P.; SRINIVASU, B. Semantic similarity between sentences. **International Research Journal of Engineering and Technology (IRJET)**, v. 4, n. 1, p. 156-161, 2017.

UFT, **Academic Regiment of Federal University of Tocantins**, 2004.

VOORHEES, E. M. Natural language processing and information retrieval. In: **International summer school on information extraction**. Springer, Berlin, Heidelberg, p. 32-48, 1999.

WINKLER, W.E. String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage, in: **Proceedings of the Section on Survey Research**, ERIC, pp.354–359, 1990.

WU, Z.; PALMER, M. Verbs Semantics and Lexical Selection, in: Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics, ACL '94, **Association for Computational Linguistics, Stroudsburg**, PA, USA, pp. 133–138, 1994.

YANG, B. **Semantic relatedness for evaluation of course equivalencies**. Ph.D. thesis, **University of Massachusetts Lowell**, 2012.

ZWEIG, M. H.; CAMPBELL, G. Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine. **Clinical chemistry**, v. 39, n. 4, p. 561-577, 1993.