Towards Concept Map Based Free Student Answer Assessment

Nabin Maharjan, Vasile Rus
Department of Computer Science/Institute for Intelligent Systems
{nmharjan,vrus}@memphis.edu

Abstract

We propose a concept map based approach to assessing freely generated student responses. The proposed approach is based on a novel automated tuple extraction system, DT-OpenIE, for automatically extracting concept maps from student responses. The DT-OpenIE system is significantly better, for assessment purposes, in terms of concept map quality than state-of-the-art open information extraction (IE) systems such as Ollie or Stanford as evidenced by our experimental results. The concept map based approach can not only generate a holistic score assessing the accuracy of a student response but also enable diagnostic feedback.

Introduction

Assessing student responses has been approached primarily using a semantic textual similarity (STS) approach in which a student response is compared to an ideal, expert-generated response.

In general, STS solutions (Agirre et al. 2015; 2016; Maharjan et al. 2017) do not explain why the two texts are similar, related or unrelated. For example, consider a question asked by DeepTutor (Rus et al. 2013), an Intelligent Tutoring System (ITS) for Newtonian Physics, and the corresponding ideal answer or expectation shown in Table 1. A student response to the question is also shown in the table.

An STS approach would most likely assign a similarity score of 3 for the given student answer meaning that the student response is missing important information. However, it does not explain which information is missing. If such explanatory functionality existed that could explain that the student is missing information about direction, an ITS could use this diagnostic information to generate a follow-up question such as: What other type of information is provided by acceleration?

One approach to add an explanatory layer in STS systems is to align text chunks, e.g., phrases, in a given pair of texts and label them with semantic relation types and similarity scores as proposed in the pilot interpretable Semantic Textual Similarity task (iSTS; (Agirre et al. 2015)).

Another approach is to use a concept map approach such as the one proposed here to both assess and interpret the student answers. A concept map is a graphical representation of organized knowledge. Concepts are the labeled nodes and relationships between concepts are the directed labeled edges of the graph. It can be a hierarchical map (Novak and Musonda 1991) where the most general concepts are at the top. The map can be also associative where no hierarchy is assumed - the concept map is a semantic network of concepts and their interrelations (Deese 1966). Since the concept maps derived from student free responses in the domain of Newtonian Physics is typically associative, we use associative concept maps in our work.

In our concept map approach, we first map ideal answers, i.e., expectations, to say, Physics problems, into ideal concept maps consisting of one or more tuples. Similarly, student responses to the same problem are mapped into corresponding concept maps. Finally, by comparing the two, we can determine whether the student answer matches one or more of the tuples in the ideal concept map and which tuples are not matched. A tuple is a triplet consisting of a relation/labeled-edge and the corresponding concepts/nodes in a concept map.

Typically, the ideal concept maps are manually created by experts from ideal answers provided by domain experts. For example, the expectation in Table 1 can be represented by a concept map consisting of two tuples: (acceleration, provides, magnitude) and (acceleration, provides, direction). On the other hand, the student concept map is automatically extracted by an open IE system from actual student responses. Ideally, a concept map with a single tuple, (acceleration, gives, magnitude), is extracted from the student response in Table 1. The result of the comparison of the two concept maps is that one tuple is missing from the student answer. We thus infer that the student answer is partially correct. Furthermore, we can provide the feedback component...
Problem: Two hockey players pass a puck between them on an ice rink. Assume that the ice is smooth so that there is no friction. What forces are acting on the puck while the puck is moving on the ice between the two players? Describe the motion of the puck.

Expectations:
1. When an object moves with constant velocity, net force on the object is zero.
2. The forces acting on the puck while it is between the players are the force of gravity and the normal force from the ice.
3. Puck moves in a straight line with a constant speed.

Table 2: A task in the DeepTutor system and its expectations.

- Problem: Two hockey players pass a puck between them on an ice rink. Assume that the ice is smooth so that there is no friction. What forces are acting on the puck while the puck is moving on the ice between the two players? Describe the motion of the puck.

- Expectations:
  1. When an object moves with constant velocity, net force on the object is zero.
  2. The forces acting on the puck while it is between the players are the force of gravity and the normal force from the ice.
  3. Puck moves in a straight line with a constant speed.

- Table 2: A task in the DeepTutor system and its expectations.
Tuples are extracted. However, given the text: "The acceleration of a system is zero, the net force is zero." many tuples are extracted including the desirable tuples (acceleration of a system is zero, force is zero). As long as the net force acting on the object is zero, the object will continue moving with constant velocity in a straight line.

DeepTutor Open Information Extraction

As mentioned earlier, state-of-the-art IE systems such as Ollie (Schmitz et al. 2012) and Stanford-OpenIE (Angeli, Premkumar, and Manning 2015) are more suited for building knowledge bases by extracting factual tuples from professionally written texts. As such, these systems do not produce desirable tuples for student assessment tasks from. To address this drawback, we propose a novel open tuple extraction method, DT-OpenIE, which is more suited for the assessment task.

Another issue with the Stanford-OpenIE tool is that its natural logic inference system tends to over-produce tuples from texts. This is helpful for solving the KBP problem where the relations from the extracted tuples are mapped onto standard KBP relations based on co-occurrence statistics, which typically requires a large amount of tuples for better estimates. However, all maximally entailed shorter clauses might not be valid for tasks such as the student answer evaluation where the focus is on whether the student has mastered specific domain concepts or not. For example, the Stanford-OpenIE tool also generates (force, is, zero) from the text T2 above, which is invalid because it is "net force" that is zero and not any individual force. Similarly, for the text "the frictional force cancels normal force", the desirable tuple output is (frictional force, cancels, normal force); however, the Stanford-OpenIE tool also generates (frictional force, cancels, force), (force, cancels, normal force) and (force, cancels, force) which are all misleading for assessment.

The Ollie system does not suffer from the problem of over-generating the tuples. Its patterns are effective at extracting tuples that mostly cover the concepts in the given short text. However, the system might retrieve false tuples sometimes. For example, the tool retrieves the incorrect tuple (mover’s push, equal, the oppose force of friction) from the text "The mover’s push equals the opposing force of friction" but fails to extract any tuple from the simpler text "The mover’s push equals the opposing force of friction".

The newly proposed extraction method called DT-OpenIE, shown in Figure 1 builds the strengths of these open IE systems and avoids their weaknesses with respect to our target task. It consists of i) a Clause Segmentation Model, ii) the Ollie System iii) DT patterns and iv) Tuple Filtering. The tuple filtering removes any duplicate tuples and produces a final concept map from the given text. Next, we describe our clause segmentation model.

<table>
<thead>
<tr>
<th>System</th>
<th>P</th>
<th>R</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>CM03</td>
<td>87.99</td>
<td>81.01</td>
<td>84.36</td>
</tr>
<tr>
<td>CMPR02</td>
<td>90.18</td>
<td>78.11</td>
<td>83.71</td>
</tr>
<tr>
<td>CM01</td>
<td>84.82</td>
<td>78.85</td>
<td>81.73</td>
</tr>
<tr>
<td>MP01 (Molina and Pla 2001)</td>
<td>70.85</td>
<td>70.51</td>
<td>70.68</td>
</tr>
<tr>
<td>DT-CS</td>
<td>81.21</td>
<td>74.25</td>
<td>77.57</td>
</tr>
</tbody>
</table>

Table 3: Results of different systems on CoNLL-2001 shared task test data. CM03 (Carreras and Marquez 2004), CMPR02 (Carreras et al. 2002), CM01 (Carreras and Marquez 2001). P = Precision, R = recall, F= F-measure.

Table 4: An optimal clause split generated from text: the speed of the desk will increase since more force is being applied.
Table 5: A list of DT patterns for tuple extraction.

<table>
<thead>
<tr>
<th>Extraction</th>
<th>Input Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>(velocity, increase, NONE)</td>
<td>NP + VP e.g. velocity increases.</td>
</tr>
<tr>
<td>(IMPERSONAL, impress, you)</td>
<td>To-clause e.g. He has ability to impress you.</td>
</tr>
<tr>
<td>(IMPERSONAL, is, no force)</td>
<td>NP₁ + VP + NP₂, NP₁ ∈ EX tag e.g. There is no force.</td>
</tr>
<tr>
<td>(1st Law, says, COMPLEX)</td>
<td>Attribution relation e.g. 1st Law says that the object moves with a constant velocity</td>
</tr>
<tr>
<td>(Push, equals, friction)</td>
<td>NP₁ + VP + NP₂, NP₁ /∈ EX tag e.g. Push equals friction</td>
</tr>
</tbody>
</table>

Clause Segmentation Model

A clause is a text segment containing a subject and a predicate. It constitutes a meaningful unit, which ideally is a proposition. Similar to Stanford-OpenIE, we extract shorter clauses from a given text and consider them candidates for tuple extraction. However, we do not use the entailment restriction or the natural logic inference system while extracting shorter clauses because of the issues discussed above.

We developed our model using the CoNLL-2001 shared task data for clause identification (Sang and D´ejean 2001). We could not replicate Adaboost classifier used by Carreras (Carreras and Márquez 2001) due to memory limitation. Instead, we used a liblinear classifier following their approach to build our model. First, we developed models to detect clause start and end boundaries and then a clause identification model that classifies whether a given clause candidate is a clause or not based on a confidence score. We extract clause candidates C(i,j) such that j > i, wordᵢ ∈ S, wordⱼ ∈ E from the text, where the wordᵢ ∈ S indicates the word at position i is tagged with a clause start label S while the wordⱼ ∈ E means that the word at j is tagged with a clause end label E.

We evaluated the clause candidates based on confidence scores to produce a clause split from the text. A clause split is a list of consistent clauses in which clauses are either nested or not overlapping. We produced a clause split from texts using both a greedy approach (Carreras and Márquez 2001) and an optimal approach (Carreras et al. 2002).

We used all but Sentence Pattern features from Carreras’ (Carreras and Márquez 2001) as they were found to be not discriminating enough for our liblinear model. Besides these features, we used some additional context features in our models. We also used some post processing rules to correct the label predicted by clause start and end classifiers. We don’t discuss them here because of space reason. Table 3 provides the performance of our clause segmentation model (DT-CS) using the optimal approach on the CoNLL-2001 shared task test data. The Adaboost algorithm with weak decision trees (CM01 and CMPR02) seemed more predictive than our liblinear model. Our system results are comparable to the top performing systems (CM01, CMPR02 and CM03). More importantly, our system extracts clauses which are reasonably suited for the student answer assessment task. Table 4 shows the clause split output for an example text using the optimal approach.

DT Patterns

We passed the output of the clause segmentation model through the Ollie system to generate the tuples. However, as discussed above, there are certain sentence structures which are not captured by the Ollie system that might have pedagogical value. Therefore, we applied a set of patterns to extract tuples from such sentence forms as listed in the Table 5. We used the special keywords IMPERSONAL and NONE to indicate the absence of first and second arguments, respectively. We used the COMPLEX keyword to denote entities which are clauses.

Experiment and Results

Data

In order to evaluate the proposed approach, we used student answer data from logged interactions of 41 high school stu-
Table 7: Mean ratings for concept maps of ideal student answers generated by different open information extraction methods. The standard deviations are provided in bracket alongside means.

<table>
<thead>
<tr>
<th>Measure</th>
<th>Stanford</th>
<th>Ollie</th>
<th>DT-OpenIE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>2.71 (1.24)</td>
<td>2.19 (1.35)</td>
<td>1.89 (1.10)</td>
</tr>
<tr>
<td>Coverage</td>
<td>2.63 (1.28)</td>
<td>2.38 (1.27)</td>
<td>1.69 (1.12)</td>
</tr>
<tr>
<td>Pedagogy</td>
<td>2.52 (1.40)</td>
<td>2.41 (1.44)</td>
<td>1.70 (1.24)</td>
</tr>
</tbody>
</table>

The results are promising - the concept maps generated by our method, on an average, fall between complete and mostly accurate for all of the three quality scales. One case where the system fails to generate tuples is a list-type student response. For example, a list-like representation for the Expectation 2 of the problem in Table 2 might be "Force of gravity and normal force". A simple resolution might be extracting such text as (force of gravity and normal force, NONE, NONE). In another approach, the tool might extract the tuple (force of gravity and normal force, ACT ON, PUCK) by inferring the missing relation and second argument (capitalized) from the dialogue context, which in this case, is a question by the DeepTutor: "What forces are acting on the puck?"

Conclusion

We presented a novel automated concept map extraction method and system, called DT-OpenIE. The experiments indicate that the generated tuples are significantly better in quality than those extracted by the state-of-the-arts open information extraction tools such as Stanford-OpenIE and Ollie systems. Our future work will focus on better tracking student’s knowledge states by using the proposed concept map approach. We also plan to exploit concept maps for dynamically providing diagnostic feedback in an automated tutoring environment and study its impact on tutoring effectiveness, i.e., on the ability of the tutoring system to induce learning gains for the learners.

Acknowledgments. This work was partially supported by The University of Memphis, the National Science Foundation (awards CISE-IIS-1822816 and CISE-ACI-1443068), and a contract from the Advanced Distributed Learning Initiative of the United States Department of Defense.
References


Ausubel, D. P. 1963. The psychology of meaningful verbal learning.


