Acquisition of Triples of Knowledge from Lecture Notes: A Natural Language Processing Approach

Thushari Atapattu, Katrina Falkner, Nickolas Falkner
School of Computer Science
University of Adelaide, Australia
(+61)883136178
{thushari.atapattu, katrina.falkner, nickolas.falkner}@adelaide.edu.au

ABSTRACT
Automated acquisition of knowledge from text has been utilised across several research areas including domain modeling of knowledge-based systems and semantic web. Primarily, knowledge is decomposed as fragments in the form of entities and relations called triples (or triplets). Although empirical studies have already been developed to extract entities (or concepts), relation extraction is still considered as a challenging task and hence, performed semi-automatically or manually in educational applications such as Intelligent Tutoring Systems. This paper presents Natural Language Processing (NLP) techniques to identify subject-verb-object (SVO) in lecture notes, supporting the creation of concept-relation-concept triple for visualisation in concept map activities. Domain experts have already been invested in producing legible slides. However, automated knowledge acquisition is challenging due to potential issues such as the use of sentence fragments, ambiguity and confusing use of idioms. Our work integrates the naturally-structured layout of presentation environments to solve semantically, syntactically missing or ambiguous elements. We evaluate our approach using a corpus of Computer Science lecture notes and discuss further uses of our technique in the educational context.

Keywords
Triples, lecture notes, relation extraction, NLP, concept map.

1. INTRODUCTION
Automated annotation of unstructured text, which is decomposed as entities and relations, is beneficial for wide variety of applications. Among them, within the educational context, knowledge-based systems such as intelligent tutoring systems benefit from semi- or fully automated domain modeling. Concept map activities such as *skeleton* maps to fill missing nodes or links benefit from adopting concept map mining (CMM) techniques as a way of reducing manual workload.

Although, previous studies focused on entity extraction [1], relation extraction is still challenging, with many techniques adopting pre-defined relations or 'named entities' (e.g. location) [2] and hence, restricted to specific domains. Although supervised learning approaches are more efficient, majority of such algorithms inapplicable to extract undefined relations. Technical disciplines like Computer Science lack named entities or pre-defined patterns and hence, not possible to reuse existing works.

This paper discusses a tool developed to automatically extract triples from lecture notes. Concept map extraction from text books is covered in other works [3]. Domain experts have already been invested in producing legible slides, allowing their expended effort to be applied to more activities that are beneficial for both the teacher and the learners. However, using NLP techniques to extract knowledge is challenging due to the noisiness of the data including use of sentence fragments, idioms and ambiguity. Therefore, we utilise contextual features such as the natural layout of presentation framework to resolve syntactically and semantically missing or ambiguous elements. This includes allocating missing subject or objects of fragments, resolving pronouns using a novel algorithm. Unlike other works which incorporate triple extraction from well-written sentences [5-6] or text books [3], to our knowledge, there are no studies until this which have implemented a full scale triple extraction from ill-written text in educational materials.

Two human experts having knowledge in Computer Science and linguistics were recruited to participate the experiments; 1. pronoun resolution 2. triple annotation. The comparison between machine and human extraction and the agreement between human experts is presented using *accuracy (F-measure)* and the *positive specific agreement* [7] respectively. We hypothesise that our proposed system is effective if *human-to-machine agreement is greater than or equal to human-to-human agreement* [8].

2. RELATED WORK
Triple extraction from Biology text books has been studied in a previous work [3] which presented a drawback of their failure to extract every triple from every sentence which leads to poor coverage of number of triples and therefore, poor pedagogical value. Triple extraction using heuristics [5] compares 3 popular parsers: Stanford/OpenNLP, link parser and Minipar. We reuse their work; however, heuristics proposed are restricted to unambiguous, complete sentences. Authors in [6] extracts all possible ordered combinations of three tokens (i.e. triple candidates) to train SVM using human annotated triples. This work has a limitation of considering all combinations of three tokens which exponentially increase with the length of sentences.

3. CONCEPT MAP MINING
Our core research focus is on automatically extracting concept maps from lecture notes to provide variety of assessment/reflective activities for learners. Initially, noise is automatically reduced using co-occurrence analysis techniques. NLP-based algorithms developed to extract concepts and rank them using structural features such as number of incoming and outgoing links, proximity and typography factors. Finally, the system produces a CXL (Concept map extensible language) file to visualise concept maps using IHMC cmap tools (http://cmap.ihmc.us/). These techniques are broadly discussed in our previous works [1,4]. The nature of presentation framework

encourages incomplete, ambiguous sentences and hence, increases the difficulty of the automated knowledge acquisition. Section 4 discusses the contextual features to solve the probable issues.

4. CONTEXTUAL FEATURES

The 'word window model' is a valid approach to solve word sense disambiguation [9]. It considers a window of *n* words to the left and right of the 'ambiguous term' to determine the context of the target word. The window can be several words in same sentence, several sentences in paragraph, or a document. By applying this method to our problem, we utilise contextual information embedded in slides to resolve ambiguity. To support our claim, we assume that the slide heading reflects the content in that particular slide. Further, we assume that each bullet-point shares logical relations with its sub points. However, there is no guarantee that an existence of logical relation between preceding and succeeding sentences in same indentation levels.

1. Syntactic rules (subject or object allocation)

This section proposes an approach to nominate syntactically missing elements in fragments. There are two main types of fragments in English called noun phrases (NP) and verb phrases (VP). Noun phrases contain a noun(s) followed by a verb. Therefore, noun phrases require an 'object' to create subject-verb-object triples. Similarly, verb phrases contain a verb followed by a noun(s) which requires 'subject' to form a complete sentence. More information on the grammatical meanings of tags can be found in http://bulba.sdsu.edu/jeanette/thesis/PennTags.html.

1. NP which contains the pattern [NP VP[VB]] look forward for candidate nouns in 'child' (i.e. sub-indentation) levels to allocate missing 'objects'.
2. VP which contains the pattern [VP[VB NP]] look backward for nouns in 'parent' (i.e. preceding-indentation) levels to allocate missing 'subjects'.

Weights are assigned to candidate nouns based on features such as grammatical structure (e.g. nouns, verbs), distance from 'input fragment' to candidate, number of tokens in the candidate phrase, whether it is an immediate level (backward or forward) or not. The weight calculation finds the subject or object to transform fragments into complete sentences.

2. Semantic rules

Lecture notes consist of semantic ambiguities such as pronouns. The widely used approach for pronoun resolution is utilising 'named entities'. Other works include searching replacement candidates in the same sentence or backward and forward search of preceding and succeeding sentences [11]. Since lecture notes lack logical relations between preceding and succeeding sentences, we propose a new algorithm.

Pronoun resolution

We applied a mechanism proposed in [11] to find replacements when bullet-point contains multiple sentences. Additionally, we find replacements in 'parent-levels', which is the preceding indentation level or heading. We assign weights for each candidate according to features such as 'location' of the candidate, distance from the pronoun, grammatical structure, grammatical number (singular or plural). The most suitable candidate is chosen using weights.

Demonstrative determiners

Lecture notes often contain demonstrative determiners (e.g. *this, these*), a word or phrase that occurs together with a word(s) to express the reference of that word(s) in the context. Our proposed approach to resolve them only considers lexical reiterations (e.g. *these calls-> system calls*). We consider features like grammatical number (singular or plural), number of strings overlaps with the candidate, grammatical structure and the determiner.

5. TRIPLE EXTRACTION

We propose a new set of features to extract entity-relation triples from English sentences. Our feature set is applicable regardless of the pre-defined patterns as in [5]. However, reusing their work might improve the accuracy in specific sentence patterns [5-6]. Our addition of new features is a consequence of broad analysis of approximately 140 lecture slide sets from different courses. This work has the potential for reuse for any knowledge source by eliminating features specific to the presentation framework.

The NLP annotation includes parsing the sentence through Stanford statistical parser [10] and link grammar parser [12]. In order to assist better understanding of the features, we derive a decision tree (Figure 1).

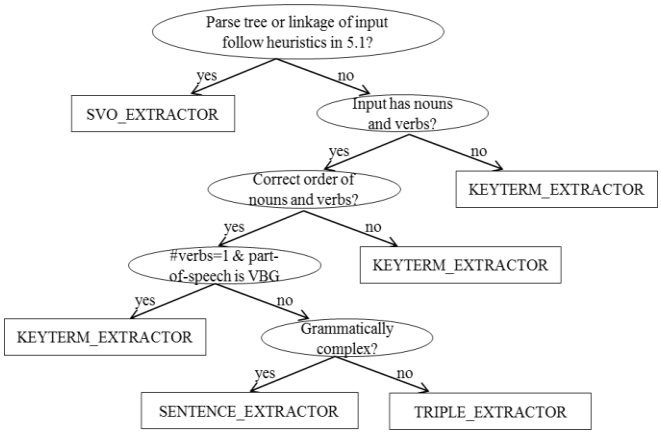

Figure 1. Decision tree which describes features and actions

1. Linguistic-based heuristics

Syntactic parse tree

Figure 2 illustrates a parse tree based on Stanford parser [10].

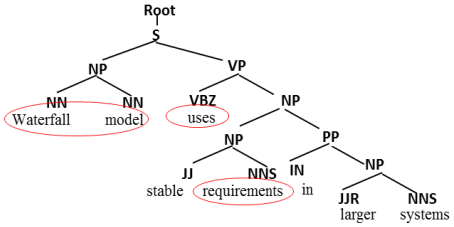

Figure 2. Parse tree of an example sentence

The following heuristics are based on previous work [5]. If the sentence contain the pattern 'Root (S (NP_subtree) (VP_subtree)), it applies following rules to extract SVO triples.

Rule 1 (subject): Perform breadth first search in NP_sub tree and select first descendant of NP_sub tree

Rule 2 (verb): Search in VP_sub tree for deepest verb descendent

Rule 3 (object): Search in PP, NP or ADJP siblings of the VP_sub tree. In NP or PP_sub tree, select first noun or compund noun, or in ADJP sub tree, select first adjective

We extended these heuristics to extract prepositional phrases.

Linkage

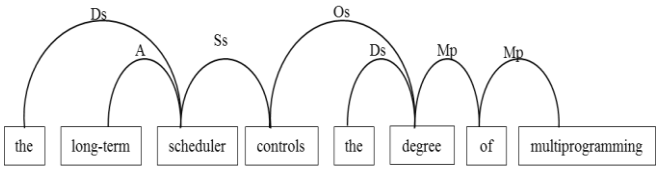

Figure 3. Linkage diagram of an example sentence

Figure 3 illustrates the linkage diagram obtained from link grammar parser [12]. Following heuristics are based on [5].

Rule 1(subject): Selects the word left of S_link

Rule 2(verb): Select first word right of S_link until {Pv, Pg, PP, I, TO, MVi} links found

Rule 3(object): Select links from 'verb' until {O, Os, Op, MVpn} links found

According to the decision tree, if input sentence follows one of the rules above, we simply extract SVO from them. However, there are many variations of sentence patterns found in lecture notes which arise us to exploit new features.

2. Sentence-based features

We extract the part-of-speech of all sentences filtered out from the criteria above. Our previous work implemented a greedy approach to identify nouns, compound nouns with their adjectives and verbs [1]. These extractions are checked against 'order' where a verb should be in-between two noun(s) to form an entity-relation triple. The candidate list should contain at least one none gerund verb. Computer Science domain contains verbs in its *–ing* form (called *gerund* -VBG), which can be used as nouns (e.g. *Software testing*). All the sentences exclude from above criteria might not produce triples and hence, important key terms are extracted from them (KEYTERM_EXTRACTION) [1].

The grammatical complexity is checked in remaining sentences using number of nested sentences (S) and dependent clauses (SBAR). If the sentence is identified as 'complex', but complete, Stanford typed dependency parser [10] splits them into simple sentences (SENTENCE_EXTRACTOR) and repeats all steps in the decision tree. The filtered out sentences consider fallback features (TRIPLE_EXTRACTION) such as number of nouns, number of verbs, numerals and symbols, negative verbs, subject-object distance, subject-verb distance, verb-object distance and headword of the sentence. We determine whether any candidate nouns are emphasised using different font colors, underline. This expresses the importance of terms to be selected as triple candidates. However, this feature is specific to the presentation framework. Finally, the extracted triples are checked against 'redundancy cycles' where the subject is repeated in an object.

6. EVALUATION

We selected lecture slide sets from recommended text books (e.g. *Software Engineering* by Sommerville) and Computer Science courses taught across different undergraduate levels in our University. We demonstrate our work using Microsoft PowerPoint, but our tool is applicable to other formats such as OpenOffice and Keynote with a structured template for header and text. Each selected lecture slide sets contains combination of contents such as text, programming, figures and notations.

Experiment 1 – Pronoun resolution

We observed that pronouns under study include *you, we, us, itself,* addressing students who refer to the course material. Due to lack of replacements in the context, we exclude these pronouns.

Table 1. Statistics of pronouns discovered in our corpus

Pronoun	they	their	it(s)	itself	we	them	you(r)	us
Frequency	57	51	241	17	23	34	94	22

Two human experts were recruited to nominate replacement candidate within a context of the slide. We did not provide replacements proposed by the system since it can influence the human judgment. We compare both of their pronoun resolution with machine's prediction and results are averaged. In table 2, *accuracy* (F-measure) is calculated as the harmonic mean between precision and recall and the *agreement* between participants is calculated using positive specific agreement [7].

Table 2. Accuracy and agreement of pronoun resolution

Lecture	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Frequency	17	16	0	67	54	39	44	50
Accuracy	0.857	0.66	-	0.746	0.923	0.587	0.5	0.571
Agreement	0.8	0.33	-	0.916	0.9	0.727	0.8	0.68

Lecture	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Frequency	5	10	7	7	40	23	4
Accuracy	0.5	0.909	0.19	0.41	0.528	0.857	0.66
Agreement	0.6	1	0.142	0.571	0.384	1	0.5

Table 2 verifies that the use of pronouns in courses vary depends on authors (e.g. L3=0). It is evident that courses which demonstrate grammatically rich, consistent writing styles provides probable replacement candidates, allowing computer algorithm to accurately (accuracy>0.8) resolve pronouns (e.g. L1- *software architecture*). As highlighted in the table 2, in some courses, accuracy is greater than human agreement. This validates our original hypothesis. We observed that the agreement is dropped when one rater suggests a replacement while other flagged it as 'null' when they find it uncertain. Occasionally, some of machine replacements did not overlap with human, reducing the accuracy as shown in L11 and L12. Dependent clauses appeared to be the main cause for this. Besides, some sentences include dummy pronouns (e.g. it is raining) which do not contain a corresponding replacement. Our results cannot be compared with other works since our corpus under study is different (i.e. lecture notes).

Experiment 2 – Triple extraction

This study uses different slide sets from experiment 1, but same Computer Science courses. We extracted 1996 sentences from 15 slide sets with approximately 40 slides per lecture note. The average number of sentence per slide is 3.3. From that, 265 sentences excluded due to 'insolvable' pronouns (highlighted in Table 1). We extracted 1838 triples from rest of the 1731 sentences. A sentence can consists of no, one or more triples.

Similar to experiment 1, two human experts participated to identify subject-verb-object triples. There is no guarantee that

human annotations are identical with machine extracted triples since our algorithm mapped sentences into their base form using lemmatisation techniques. Therefore, we calculated string similarity between each subject, verb and object and obtained an average score. An example of similarity calculation between subjects is shown below and more details can be found in [6].

- Computer (sub) – *drawback of waterfall model* (tokens=4)
- Human (sub`) – *waterfall model* (tokens=2)

Sim (sub, sub`) = overlap / (# tokens in x; x= max(sub, sub`))
= 2/4 = 0.5

Verb and object similarity is calculated in the same way. The final similarity between computer and human is ranged between 0-1, stressing 1 is identical and 0 means no overlap. We measured the *precision* by comparing computer extracted triples to human and *recall* when performing the other way around and obtained the mean using F-measure (accuracy).

Table 3. Accuracy and agreement of triple extraction

Lecture	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
# Triples	107	81	24	173	207	108	145	221
Accuracy	0.862	0.507	1	0.605	0.872	0.397	0.88	0.944
Agreement	0.928	0.808	1	0.761	0.930	0.623	0.8804	0.975

Lecture	**9**	**10**	**11**	**12**	**13**	**14**	**15**
# Triples	82	134	74	180	130	110	62
Accuracy	0.787	0.833	0.465	0.319	0.497	0.858	0.672
Agreement	0.829	0.792	0.66	0.645	0.72	0.844	0.76

According to Table 3, it is evident that some courses produces acceptable machine performance (accuracy>0.8) (e.g. L8-*Software engineering*). Computer networking slides (L3) from text book (source can be found in http://williamstallings.com/DCC6e.html) achieved an accuracy of 1, resulting in an ideal machine extraction. The accuracy is varying based on the richness of the content. Our algorithm is more effective (accuracy>0.8) for courses categorised as Software engineering, computer architecture, communications (see ACM classification in http://en.wikipedia.org/wiki/Outline_of_computer_science). We recognise these contents are *well-fitted* (e.g. rich grammar, complete sentences with apparent independent clauses) for CMM. Other courses with combinations of good text and notations (e.g. L15-distributed systems) are categorised as *average-fitted* (accuracy>0.5). The courses with low accuracy (<0.5) (e.g. programming languages, data structures) are classified as *ill-fitted* (More information on the classification can be found in [4])

Our results show that accuracy is greater than or equal to inter-rater agreement in some courses which validates our original hypothesis into some extent. The agreement varies when one party (computer or human) extracts modifiers while the others extracts only the exact words. The machine performance is dropped in some occasions (e.g. L6, L11 and L12) mainly due to our failure to handle negations correctly. It is practically challenging for machine to outperform human in a corpus like lecture notes since there is no well-defined structure for writing course materials.

An important aspect of studying concept maps mined from lecture notes is to facilitate students in understanding relationships between concepts allowing effective knowledge organisation which is not supported in the linear nature of lecture notes. The aim of this research is to adapt concept maps according to the learners' problem solving context. In future works, we evaluate our work by measuring students' performance in given tasks while learning through task-adapted concept maps. Besides, CMM techniques support wider concept mapping activities such as providing scaffolding aid and domain modeling of ITS.

7. CONCLUSION

This paper proposed a novel set of features to automatically extract entity-relation triple from lecture notes. While slides may have many potential issues, including incomplete, ambiguous sentences, we introduced a novel approach to resolve syntactically and semantically missing or ambiguous elements using contextual information of the slides. Our results showed that for *well-fitted* courses, machine performance is closer to human predictions (accuracy>0.8). However, our system indicates low accuracy for *ill-fitted* contents such as programming which are undesirable for CMM. The work presented in this paper is restricted to a corpus of Computer Science courses. We plan to conduct cross-disciplinary study to observe the validity of our approach.

8. REFERENCES

[1] Atapattu, T., Falkner, K. and Falkner, N. 2012. Automated extraction of semantic concepts from semi-structured data: supporting computer-based education through analysis of lecture notes. In *proceedings of the 23rd International conference on Database and Expert systems applications*.

[2] Cunningham, H. et al. 2002. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th anniversary meeting of the association for Computational Linguistics.*

[3] Olney, A. et al. 2011. Generating concept map exercises from textbooks. In *Proceedings of the 6th workshop on innovative use of NLP for building educational applications*.

[4] Atapattu, T., Falkner, K. and Falkner, N. 2014. Evaluation of concept importance in concept maps mined from lecture notes: computer vs human. In *proceedings of the 6th International conference on computer supported education.*

[5] Rusu, D. et al. 2007. Triplet extraction from sentences. In *Data mining and data warehouses*.

[6] Dali, L. et al. 2009. Triplet extraction from sentences using SVM. In *Data Mining and Data Warehouses (SiKDD)*.

[7] Hripcsak, G. et al. 2005. Agreement, the F-measure, and reliability in information retrieval. In *Journal of the American medical informatics association*, 12(3), 296-298.

[8] Hearst, M. 2000. The debate on automated essay grading. In *Intelligent systems and their applications*.

[9] Ide, N. and Veronis, J. 1998. Introduction to the special issue on word sense disambiguation: the state of art. In *Computer Linguistic Journal – Special Issue on word sense disambiguation.* 24(1), 2-40

[10] Klein, D. and Manning, C. 2003. Accurate Unlexicalized parsing. In *Proceedings of the 41st meeting of the association for computational linguistics*, 423-430.

[11] Mitkov, R. 1998. Robust pronoun resolution with limited knowledge. In *Proceedings of the 17th International conference on Computational Linguistics*, 869-875.

[12] Sleator, D. et al. 1993. Parsing English with a links grammar. In *third International workshop on parsing technologies*.