An experimental investigation of semi-automatic generation of concept maps from textbooks

by

Vineet Vilas Nayak

A thesis submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Auburn, Alabama
August 3, 2019

Keywords: E-TextBook, Concept Mapping, NLP, Machine Learning

Approved by

Hari Narayanan, Chair, Professor of Computer Science and Software Engineering
Saad Biaz, Professor of Computer Science and Software Engineering
Debswapna Bhattacharya, Assistant Professor of Computer Science and Software Engineering

Abstract

A concept map is a useful tool to organize and structure unstructured data, in this case, the information contained in a textbook. In education, a concept map can be used to set learning goals, monitor progress and visualize relationships between concepts. Constructing concept maps manually is a complex and time-consuming task which usually requires domain experts. Automatic creation of concept maps from documents is called concept map mining. In this study, we explore various technical approaches to semi automate the process of generating a concept map from an e-textbook. The approach selected is to use the topics listed in the appendix as concepts and mine the relation between pairs of topics using natural language processing and machine learning. Limitations of this approach and directions for future research are discussed.

Acknowledgments

I would like to take this opportunity to express my gratitude to my supervisor, Dr. Hari Narayanan, for his support, excellent guidance, and providing me with a research atmosphere that challenged me and pushed me forward, and to my committee members: Dr. Debswapna Bhattacharya and Dr. Saad Biaz.

I would also like to thank my friends who supported me, stood by me, and helped me in various aspects of my life while I worked on my thesis.

Finally, I would like to thank my parents to whom I owe my achievements to, who encouraged me, provided me with everything I need, and kept me in their prayers. I would not have been able to finish my thesis without them.

Table of Contents

List of Tables

List of Illustrations

List of Abbreviations

CM	Concept map
CMM	Concept map mining
CSV	Comma separated values
NLP	Natural language processing
POS	Parts of Speech

Chapter 1

Introduction

With digital devices like phones, tablets, and computers being ubiquitous in classrooms, digital editions of textbooks are also becoming common. While most current e-textbooks are identical to the printed versions, a few have additional features such as hyperlinks to outside sources and audio-visual embedded media. Digital textbooks are also often cheaper as there no physical production costs. They are also more accessible because libraries can provide unlimited copies of an e-book.

Traditional organization of a printed text book is linear. The author of the text book decides the best order in which the reader should consume the subject matter. A traditional text book starts with table of contents, then the subject matter which is divided into chapters and finally an appendix with the important key topics and their page numbers, called an index. The book's author determines how the topics covered in the book are grouped into chapters, and some topics may be fragmented into several chapters. For example, in Figure 1 topics A, B, C and D are the topics covered in a book and A1, A2, A3… are parts of topic A, B1, B2, B3… are parts of topic B, etc. A traditional book covering these topics may then present the topics and subtopics as shown in Figure 1:

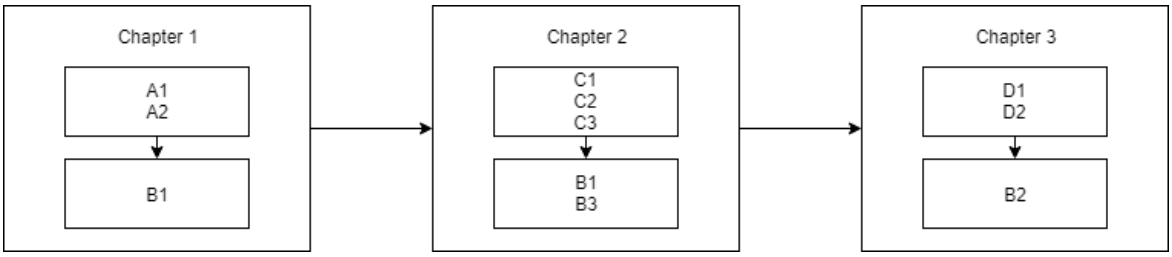

Figure 1: Structure of traditional textbook

A concept map is a graph where the nodes depict concepts or topics and the edges which connect the nodes represent the relationships between them. A concept map is a way of representing relationships between ideas, images, or words in the same way that a sentence diagram represents the grammar of a sentence, a road map represents the locations of highways and towns, or a circuit diagram represents the workings of an electrical appliance [1]. It is a useful tool to organize and structure unstructured data, in this case, a textbook.

In education, a concept maps can be used to set learning goals, monitor progress and visualize relationships between concepts. Jennifer Turns et al demonstrate how concept maps represent an innovative way to assess and gain insight, into student learning about the relationships among concepts. [2] They also describe how concept maps are useful not only as a tool for teachers trying to convey the relationship between concepts but also as a tool to assess student's understanding of relationships between concepts. Concept maps also affect how much and how long students retain information. Information learned via rote learning is quickly forgotten. However, concept maps help students learn concepts in a more meaningful fashion allowing for better understanding and longer retention. [3]

In a concept map-based textbook, instead of the subject matter being divided into chapters, the subject matter is divided into much smaller topics or concepts. These concepts are connected by their inter-relationships to form the concept map that is graphically presented. One may think

of this graph, the concept map, replacing a traditional table of contents. Clicking on the nodes or links of the graph takes the reader to multimodal (text, images, video, etc.) descriptions of the corresponding concepts and relationships. The reader may start at any point and then move on to related topics. Depending on how finely the topics are divided the map may be small and simple or large and detailed.

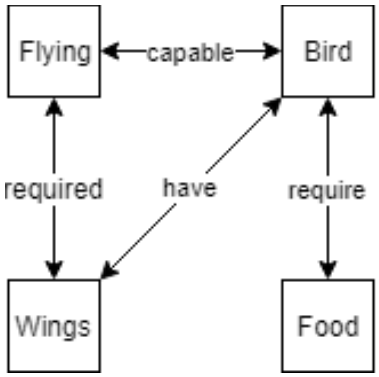

Figure 2: Simple concept map

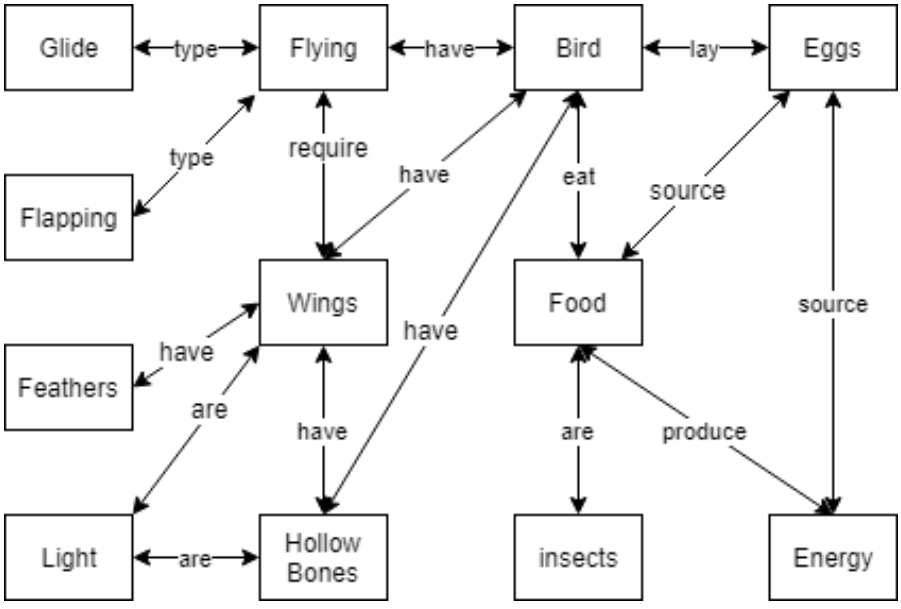

Figure 3: Detailed concept map

Constructing concept maps manually is a complex and time-consuming task which usually requires domain experts. Historically concept maps were built using Post-its™ on a whiteboard. Post-its™ allowed the creator to move around concepts easily. Concept maps are never really complete, but once the creator is satisfied, they could be copied down into a more permanent media like a notebook till they decide to make further revisions. [3]

Unlike a textbook, concepts maps capture and present the structure as well as content of knowledge, i.e., concepts and how they are interconnected. Therefore, concepts maps are considered to be non-linear representations that better support student learning than linear text [3]. Concept maps, therefore, are also used by educators and educational researchers to assess the knowledge of students. For example, Figure 4 shows the concept map an honors class student drew to illustrate what they had learned regarding chemistry. [4]

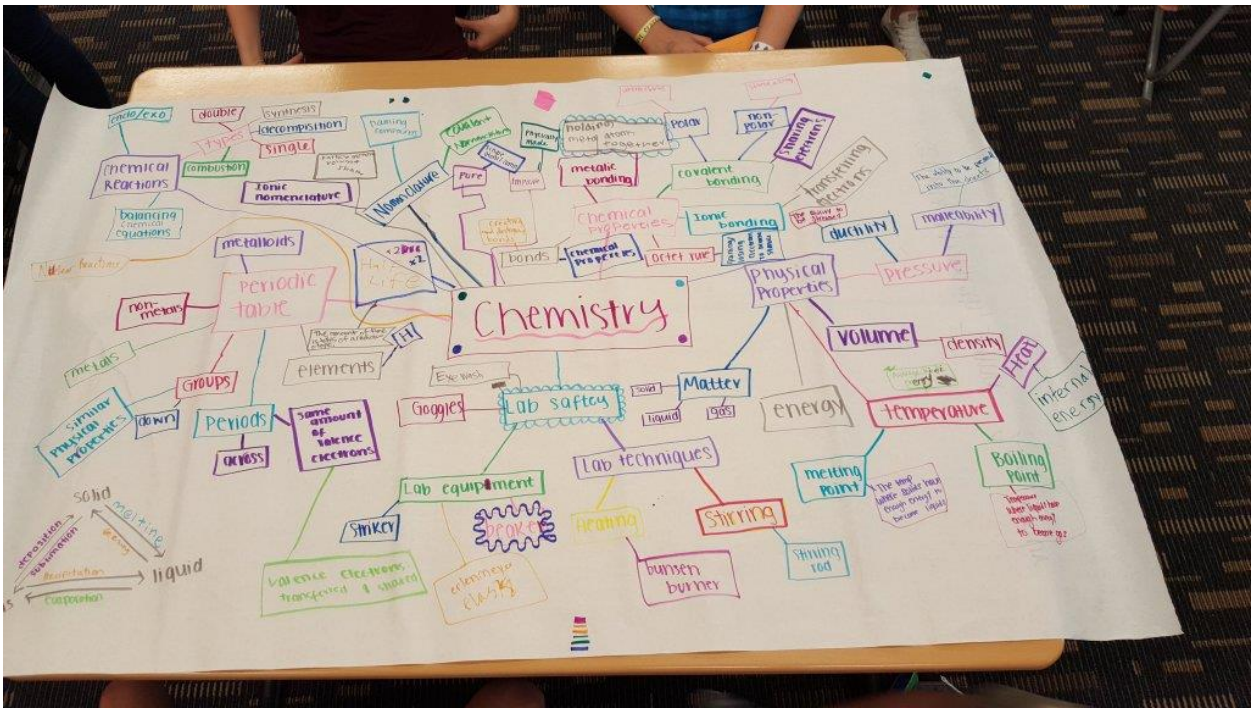

Figure 4: Concept map drawn by a student.

Automatic creation of concept maps from documents is called concept map mining (CMM). There are systems proposed for concept map mining that extract all the nouns in a text. The concepts are then chosen from this list of nouns based on statistics. [5] Two studies, by Clariana et al. [6], and by Richardson and Fox [7], describe the construction of concept maps, but their maps do not include labeled relationships.

For concept map mining we must first identify the concepts. When we rely solely on statistical analyses of text to identify concepts, the concept maps generated may not be meaningful to humans. This is one of the reasons domain experts are required to generate concept maps. Another reason why domain experts are required is identification of relationships. The relations that are mined may not be correct or maybe irrelevant. Also, since concept maps are subjective and hence can never be deemed complete or correct a domain expert will be needed to prune or simplify the concept map by removing vague nodes or weak relations.

In the case of textbooks, we already have a list of human-curated concepts in the form of an index. Hence, we can use the items listed in the appendix as concepts. Once we have a list of concepts the task then becomes to find the relations between all these concepts.

We begin by selecting two concepts from an index and finding their relation. Repeating this process will give us the relations between all pairs of concepts. First, we need to find all references to the selected topics in the textbook. To derive the relationship between any two topics we will first extract sentences containing both topics. In linguistics, coreference, sometimes written co-reference, occurs when two or more expressions in a text refer to the same person or thing i.e they have the same referent. In the sentence “Bill said he would come” the proper noun Bill and the pronoun he refers to the same person, namely to Bill. [8] Coreference resolution is the task of

finding all expressions that refer to the same entity in a text and replacing the expressions with the entity.

Once we have resolved the references in the text, we extract all sentences that mention the two concepts. We then have several sentences that describe the two topics or concepts whose relation we want to find. These sentences are complex and difficult to analyze. Hence, we will simplify the sentences using Natural Language Processing (NLP) to extract a sentence fragment that is most relevant. This sentence fragment must also show the relationship between our two selected concepts which is our goal.

A dependency parser analyzes the grammatical structure of a sentence, establishing relationships between "head" words and words which modify those heads. Figure 5 below shows a dependency parse of a short sentence. The arrow from the word moving to the word faster indicates that faster modifies moving, and the label advmod assigned to the arrow describes the exact nature of the dependency. [9]

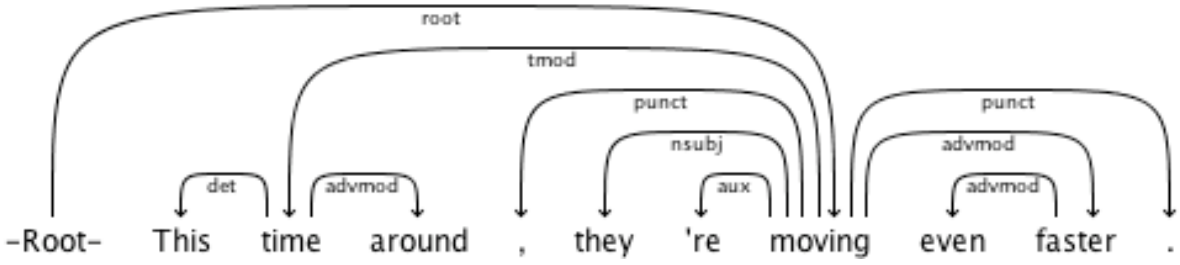

Figure 5: Sample dependency parsed tree

Using this notion of dependency parsing to generate dependency trees, we find the path between two concept nodes to generate our required sentence fragment. Not all fragments generated show the relationship between the two concepts despite containing both concepts. We use a machine learning approach to classify and identify the fragments that are more or less grammatically correct and explain the relation correctly.

In this way, by repeating the process for all pairs of concepts, we can generate a matrix of all possible relations between the topics listed in the appendix. This can then be used in the creation of the concept map. In Chapter 2, we review relevant literature. Chapter 3 describes the various experimental approaches we developed, implemented and tested in order to be able to extract the relationships between any two topics or concepts in an e-textbook, as well as details of the chosen approach (discussed at an abstract level in the paragraphs above). Chapter 4 provides results of testing this approach, and its limitations and future research directions appear in Chapter 5.

Chapter 2

Literature review

Technology has advanced to the point that machines can interpret and answer contextual questions. An area which can benefit significantly from this are e-books. However, despite digital devices such as smartphones being ubiquitous and having enough power to process queries e-books are generally still implemented as digitized versions of the printed textbooks with interactive elements such as video and animations embedded into them. In this research we propose a method of parsing the knowledge in e-textbooks into a structured format known as concept maps that is more machine readable as well as understandable to humans and allows readers to navigate through the book in a non-linear fashion. Concepts maps depict knowledge as graphs where nodes are the topics and the edges show how the topics are related to each other. In this chapter we discuss a few selected papers on the creation and mining of concept maps.

Joseph D. Novak and Alberto J. Cañas explain in their paper [3] that for construction of concept map it is important to begin with a domain of knowledge that is very familiar to the person constructing the map i.e. a domain expert. This domain expert selects what knowledge would be included in the domain. They also suggest selection of domain around a problem which they call a Focus Question. The next step is to identify and rank concepts that apply to the domain. The step after that is to construct a preliminary concept map. Historically concept maps were built by hand using Post-its ™ on a whiteboard where the Post-its ™ were stuck on the board and then connecting lines were drawn. They also explain that it is important to recognize that a concept map is never finished. The map is then revised till the expert is satisfied.

Jorge J. Villalon defines the automatic extraction of CMs from text as “Concept Map Mining” (CMM) in his paper “Concept Map Mining: A definition and a framework for its evaluation”. [10] In this paper Villalon formally defines a CM as a triplet CM = {C, R, G} where C is a set of concepts C = {c1, c2, ..., cn}, R is a set of relations between concepts R = {r1, r2, ..., rk} and G = g1, g2, ..., gm is a sorted set of generalization levels. Each concept ci corresponds to a word, or phrase, and it is unique in C. Each relation ri , is a triplet of the form ri = (cp, cq, li), where cpand cq are concepts from C, and li is the label for the relation ri which also corresponds to a word or phrase. Each generalization level gi corresponds to a set of concepts gi = c1, c2, ..., cs that share the same level of generalization.

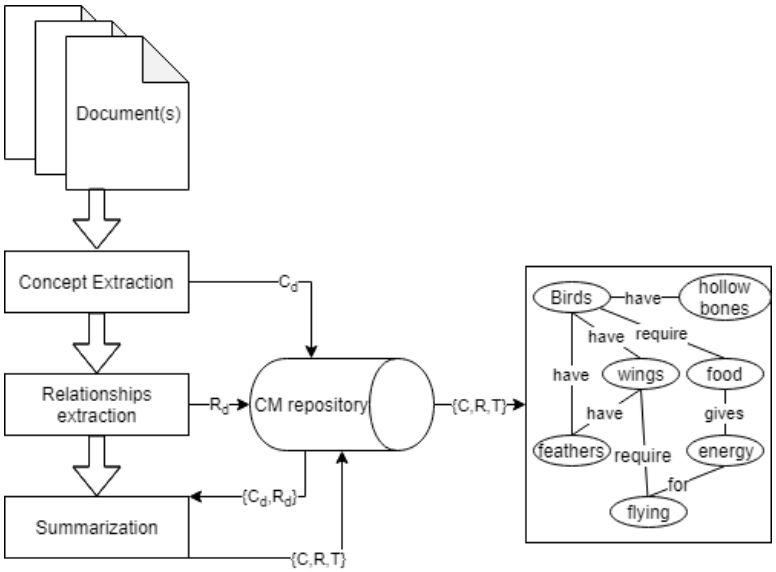

Figure 6: CMM process

They suggest a the CMM process implementation must have three modules i.e. concept identification, relationship identification and summarization. Concept identification is done using grammar trees. Nouns, verbs and adjectives, are parts of speech in written discourse, and can be identified using part of speech (POS) tagging systems. For cascading relationship identification, they propose the use of regular expressions on the grammar tree or dependencies (grammatical

relations) between concepts and the verb in between the concepts using a grammar tree. For concept and relationship summarization they suggest using Singular Value Decomposition on a term by sentence matrix from the essay, with each singular vector representing a topic sorted by the explained variance.

Krunoslav Žubrinic´ describes methods commonly used in a CMM process for unstructured texts in natural languages. [11] The source document(s) can be either single or multiple documents. He identifies that the goal of CMM studies is to produce a starting CM model which can be refined later manually. The CM generated by CMM maybe fully completed and contain concepts connected with labeled relationships or with connected concepts, but without labeled relationships, or extracted only concepts. The four approaches mentioned in his paper are the statistical approach, machine learning, usage of a dictionary, and usage of linguistics tools and techniques. Statistical methods analyze the term frequency and co-occurrences and tend to be efficient and transportable. Statistical methods may be imprecise because the semantics of terms are not considered. On the other hand, supervised or unsupervised machine learning methods are used to create rules. For the concept and relationship extraction models using these learned rules are implemented. In this process techniques such as classification, association rules, and clustering are used. In the usage of a dictionary approach for the extraction phase ontology and lists of predefined terms are used as the seed to define concepts and relationships more precisely. Across a collection of documents, it is possible to fetch terms and relationships that most frequently occur together with a particular term. For linguistics tools and techniques, statistical and data mining approaches are utilized along with lexical or semantic elements as additional features in calculations. Numerical techniques make better predictions if all similar but slightly different expressions are considered as one.

Chapter 3

Experimental Approaches

3.1 Initial Steps

Our goal was to develop a semi-automatic method to convert a digital textbook into a concept map. The first step of any CMM process is to extract concepts. Since we did not want to rely on statistical techniques and wanted to reduce the requirement of domain expertise we decided to use the appendix of the textbook as the source for concepts. The appendix of a textbook is a list containing entries with one to three words each. These entries are all concepts that a domain expert has already determined one would look up while studying the subject. Therefore, we used this list as our list of concepts. Once we have a list of concepts the next step is determining relations among all these concepts. If we can find the relation between one pair of concepts we could repeat the process to find the relations (if any) between all the pairs.

As source documents, we choose science textbooks as they would have a well-defined appendix. After looking through Project Gutenberg [12] and OpenStax [13], we selected Biology, Concepts of Biology and College Physics from OpenStax as three textbooks to be used for training and testing. However, the raw text couldn't be used as is. The textbook was converted from pdf to txt format. Then the page footers and page numbers were removed. Finally, everything before the first chapter and after the last chapter was trimmed i.e. the preface, index and appendix were removed. The appendix was stored in a separate file. The appendix file was also processed such that each appendix entry was on a new line and the page numbers were removed.

3.2 Summarization Approach

The initial approach was that we tried to find paragraphs that had both concepts and use summarization to reduce it to a single sentence and then extract a relevant sentence fragment that would give us the relationship between the two concepts. Following were the four steps involved.

1. Perform coreference resolution on the textbook.
2. Extraction of paragraphs containing the selected pair of concepts.
3. Extract key sentence(s) using summarization.
4. Extract relationship fragment using tokenization/tagging.

Most concepts in the concept map are nouns. A problem that we come across when trying to extract sentences containing said concepts from raw text is that they may be referred via expressions like pronouns. Coreference occurs when multiple expressions in a text refer to the same entity. For example, consider the sentence “Jack said he would sing.” In this sentence the proper noun Jack and the pronoun he refers to is the same person i.e. Jack. There are four main types of coreference i.e. anaphora, cataphora, split antecedents and co-referring noun phrases. Without proper coreference resolution, much of the semantic information from a text can be lost, resulting in an incomplete concept map where relationships between concepts are not found.

For coreference resolution, the following software were researched:

1. Stanford Deterministic Coreference Resolution System [14] - This java-based system developed by the Stanford natural language processing group implements the multi-pass sieve coreference resolution (or anaphora resolution) system
2. Cort [15] - Cort is short for 'coreference resolution toolkit'. It is a python-based system developed by Sebastian Martschat and Michael Strube.

3. Neuralcoref [16]- Neuralcore is a python-based system developed by Thomas Wolf at Huggingface Inc. It is a pipeline extension for spaCy 2.1+ which annotates and resolves coreference clusters using a neural network. NeuralCoref is production-ready, integrated in spaCy's NLP pipeline and extensible to new training datasets.

For this project NeuralCoref was selected for implementation. Following are a few examples of coreference resolution:

Before	After
The bear was so white that *it* couldn't be seen.	The bear was so white that *the bear* couldn't be seen.
If *they* are hungry *the bears* will eat the deer.	If *the bears* are hungry the bears will eat the deer.
Bears and tigers are both predators. *They* will fight one another if confronted.	Bears and tigers are both predators. *Bears and tigers* will fight one another if confronted.

Table 1: Example NeuralCoref input and output

For extraction of paragraphs containing the selected pair of concepts, regular expression matches were performed on all the paragraphs in the cleaned textbook. After segregation of the required paragraphs, we investigated available summarization tools and techniques. Following tools were tested:

1. Textsum [17] - Opensource tool based on Google TensorFlow
2. Open Text Summarizer [18]- Opensource tool by Nadav Rotem.
3. Resoomer [19]- Closed source online tool with paid API access.
4. Smmry [20]- Another closed source online tool with paid API access.

Following research articles on automatic text summarization were reviewed:

1. Text Summarization Techniques: A Brief Survey [21] - This article describes two types of extractive summarization i.e. topic representation and indicator representation.
2. Resource Lean and Portable Automatic Text Summarization [22] - Martin Hassel explains extractive summarization method using a genetic algorithm by selecting sentences and generating candidate summaries then executing the hill climbing algorithm by comparing the candidate's score with original text's score to see how much information has been retained.
3. A Review Paper on Text Summarization [23] - Comparative study of various abstractive and extractive text summarization methods
4. A Review on Automatic Text Summarization Approaches [24] - A study describing sentence extraction via Frequency Based Approach, Feature-Based Approach, and Machine Learning Approach.
5. Automatic Text Summarization [25] -This paper describes an approach using a genetic algorithm with mathematical regression for summarization

Summarization software and methods were all geared to extract a small paragraph from a much larger text. Most didn't accept any bias towards certain keywords. Thus, when provided with a compilation of paragraphs containing our selected concept keywords the output was a group of sentences that didn't necessarily contain one or both concepts. Even when provided with a paragraph made up solely of sentences containing the keywords the software failed much of the time to extract a key sentence that contained the relation between the two concepts. Thus, after reviewing articles and testing solutions available we concluded that current summarization tools would not provide us with the required output and a different approach would be required.

3.2 Machine Learning Approach

After summarization, the next approach we investigated was the use of machine learning algorithms to classify sentences, which could provide us the relationship between concepts. The aim was to use Naive Bayes, Logistic Regression, Support Vector Machine, Decision trees and other algorithms from prebuilt libraries to achieve this. For this purpose, the following machine learning software were considered: Splunk MLTK [26], Scikit-learn [27], Weka [28], Keras [29], TensorFlow [30] and LightSide [31].

Splunk provided a free trial to their 'Machine Learning Toolkit' platform (Splunk MLTK). However, we did not find this platform easy to use and would have required us to take part in their training program. TensorFlow although free to use would require extensive programming to implement machine learning algorithms. Keras which is an API that runs on top of TensorFlow only provided deep learning algorithms and did not provide feature extraction. Scikit-learn is another open source machine learning platform. It did provide most of what we were looking for but wasn't easy to use. Weka is a collection of machine learning algorithms for data mining tasks. However, LightSide included Weka's algorithm implementation as one of the options. LightSide is an open-source platform which includes a machine-learning and feature-extraction core as well as a user interface called researcher's workbench. Since LightSide was free, easy to use and modular, after some testing, LightSide was selected as the tool to work with.

We developed the following approach with machine learning for concept map mining:

1. Perform coreference resolution on the textbook.
2. Extract sentences containing both concepts from the coreference resolved book.
3. Use machine learning to classify the sentences that express relationship between the selected pair of concepts.

4. Extract the sentence fragment that best expresses the relationship between the concepts.

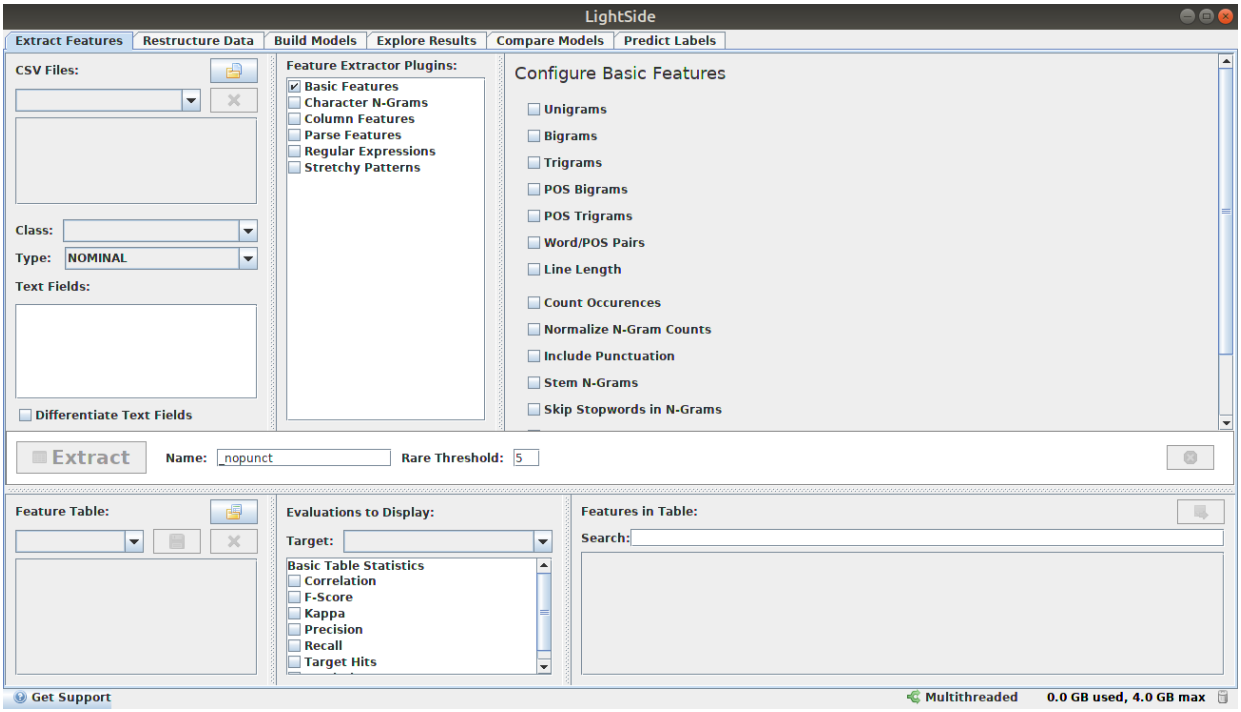

Figure 7: Feature extraction in LightSide

LightSide [31] is a tool that accepts CSV files i.e. comma separated values as input containing class predictor, sentence and features. The class predictor is manually curated as ‘pos’ or ‘neg’ in the training set and the tool generates this in the testing set. Natural Language Tool Kit (NLTK) [32] was implemented for tokenizing and tagging to generate Parts-of-Speech (POS) features to be used as input. The input was in the format <class, text, key1, key2, POS, count_NOUN, count_ADP, count_ADJ, count_VERB, count_ADV, length> where the 'class' field could contain either value 'pos' if sentence is a good candidate for relation extraction and 'neg' if it is not a viable candidate. The 'text' field contained the sentence. The 'key1' and 'key2' fields contained the two concept phrases. In the 'POS' field each word was replaced with the corresponding part of speech. The 'count_NOUN', 'count_ADP', 'count_ADJ', 'count_VERB' and 'count_ADV' each held the number of nouns, adposition, adjectives, verbs and adverbs in the

sentence. The 'length' field contained the number of words in the sentence. In the POS field, some patterns such as "Noun Verb Noun" or "Noun Verb Adjective Noun" seemed to correlate with positive sentences. So we checked for these patterns and the true/false field to denote if a pattern was found as a feature.

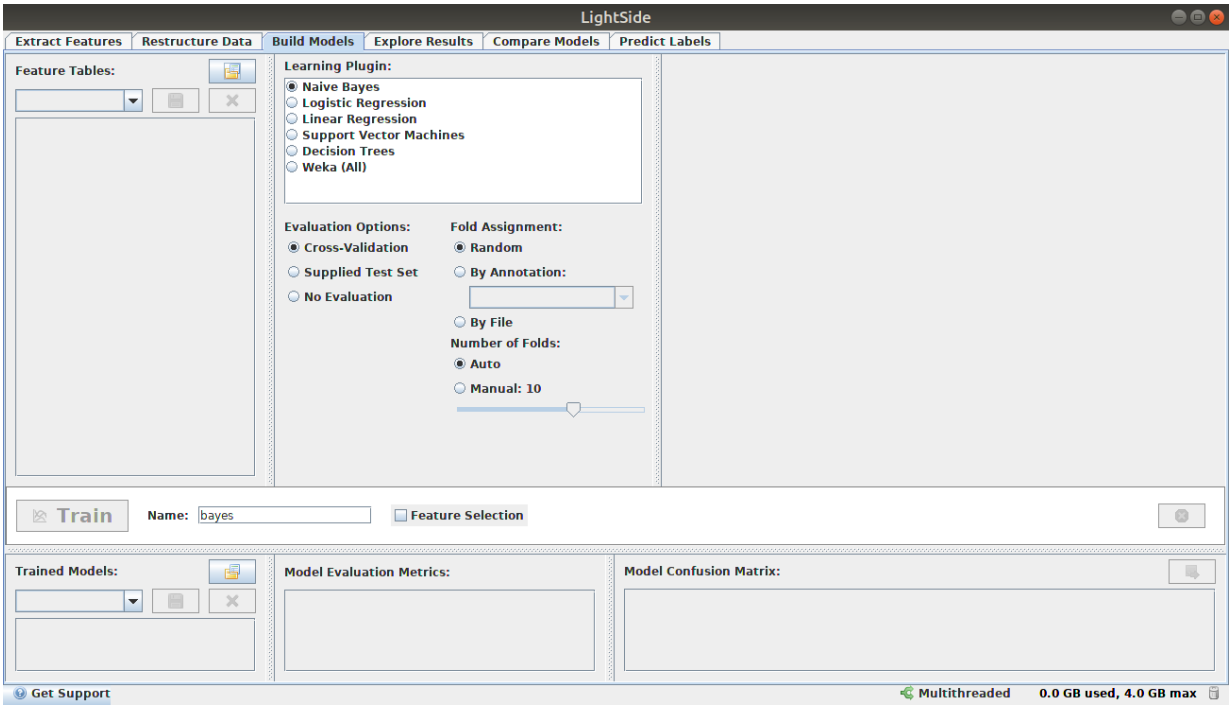

Figure 8: Model Generation in LightSide

In addition to these, the tool allows you to generate some features automatically such as unigram, bigram, trigram, POS Bigram, POS Trigram, line length, punctuation, word position pairs, etc. We used the Naive Bayes, logistic regression, support vector machines and decision trees algorithms available in the tool. However, despite several permutations of all these features, the resulting predictions were close to random. We determined that this was due to the size of the input sentences and that the sentences contained too many words unrelated to the concepts and the relation we needed, and this skewed the results. Thus, we needed to modify our approach.

3.3 Dependency Parsing with Machine Learning Approach

The aim was to reduce the sentence length so that the machine learning algorithms could accurately judge whether a sentence positively conveyed the relationship between the two concepts. While adding additional features using NLP to the input during the previous approach we came across TextRazor [33]. This online tool provided various types of analysis of a sentence using NLP including dependency parsing. Dependency parsing forms a tree around the main verb in the sentence as seen in the following figure.

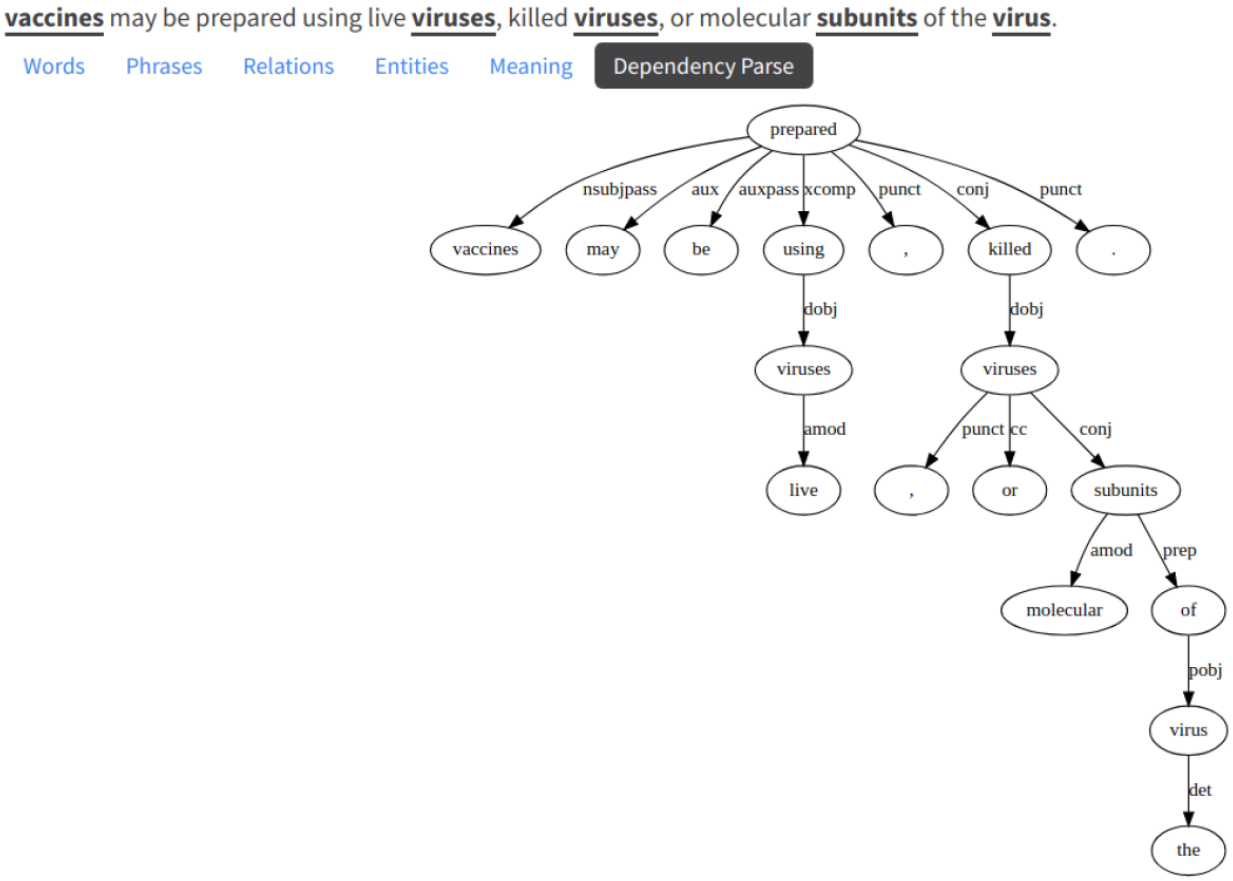

Figure 9: Dependency Parsing using TextRazor.

In the above example ‘vaccines’ and ‘viruses’ are our keywords or concepts. We are trying to determine the relation between vaccines and viruses. If we follow the shortest path from vaccines to viruses in the tree, we get ‘vaccines prepared using viruses’. This is the relation between the concepts we are looking for. Let us call this extracted part of the sentence the dependency fragment. This dependency fragment substitutes the actual sentence during feature extraction. TextRazor is however an online, proprietary and paid tool.

```
import spacy

nlp = spacy.load("en_core_web_sm")
doc = nlp(u"Vaccines may be prepared using live viruses, killed the viruses, or molecular subunits of the virus.")
for token in doc:
    print("Word=",token.text,";Dependency=", token.dep_,";Parent=", token.head.text, ";POS=",token.pos_,";Children=",
          [child for child in token.children])
```

RUN

```
Word= Vaccines ;Dependency= nsubjpass ;Parent= prepared ;POS= NOUN ;Children= []
Word= may ;Dependency= aux ;Parent= prepared ;POS= VERB ;Children= []
Word= be ;Dependency= auxpass ;Parent= prepared ;POS= VERB ;Children= []
Word= prepared ;Dependency= ROOT ;Parent= prepared ;POS= VERB ;Children= [Vaccines, may, be, using, ,, killed, .]
Word= using ;Dependency= xcomp ;Parent= prepared ;POS= VERB ;Children= [viruses]
Word= live ;Dependency= amod ;Parent= viruses ;POS= ADJ ;Children= []
Word= viruses ;Dependency= dobj ;Parent= using ;POS= NOUN ;Children= [live]
Word= , ;Dependency= punct ;Parent= prepared ;POS= PUNCT ;Children= []
Word= killed ;Dependency= conj ;Parent= prepared ;POS= VERB ;Children= [viruses]
Word= the ;Dependency= det ;Parent= viruses ;POS= DET ;Children= []
Word= viruses ;Dependency= dobj ;Parent= killed ;POS= NOUN ;Children= [the, ,, or, subunits]
Word= , ;Dependency= punct ;Parent= viruses ;POS= PUNCT ;Children= []
Word= or ;Dependency= cc ;Parent= viruses ;POS= CCONJ ;Children= []
Word= molecular ;Dependency= amod ;Parent= subunits ;POS= ADJ ;Children= []
Word= subunits ;Dependency= conj ;Parent= viruses ;POS= NOUN ;Children= [molecular, of]
Word= of ;Dependency= prep ;Parent= subunits ;POS= ADP ;Children= [virus]
Word= the ;Dependency= det ;Parent= virus ;POS= DET ;Children= []
Word= virus ;Dependency= pobj ;Parent= of ;POS= NOUN ;Children= [the]
Word= . ;Dependency= punct ;Parent= prepared ;POS= PUNCT ;Children= []
```

Figure 10: Dependency Parsing and POS tagging using spaCy.

spaCy [34] is a free open-source library for Natural Language Processing in Python. It features named entity recognition (NER), parts of speech (POS) tagging, dependency parsing, word vectors and more. The NLTK POS tagging used in the previous approach was replaced with spaCy and dependency parsing was implemented.

Following was the approach we developed for concept map mining using dependency parsing with machine learning:

1. Perform coreference resolution on the textbook.
2. Extract sentences containing both concepts from the coreference resolved book.
3. Extract sentence fragments using dependency tree and by finding the shortest path between the two concept nodes.
4. Use machine learning on the dependency fragments to classify the fragments and retain only those that correctly define the relationship between the concept pair.

With sentence fragments as input, the outputs of machine learning algorithm improved. Few features like the regular expressions that were meant for complete sentences were dropped. The syntactic functions i.e. the edges of the dependency tree were added as a feature under 'fragment_dep'. 'fragment_POS' similar to 'POS' is the words in the fragment replaced by the corresponding POS. The new format for the comma-separated values (CSV) became <class, dependencyFragment, text, key1, key2, fragment_dep, fragment_POS, count_NOUN, count_ADP, count_ADJ, count_VERB, count_ADV>. We created a test sample of around 5000 sentences extracted across the textbooks Biology-OP, Concepts of Biology-OP and College Physics and manually classified as positive (shows the relationship between concept pair) and negative (doesn't show relation). The training data overwhelmingly contained negative sentences, the ratio of positive to negative sentences was approximately 1:10. As a result a failed model would

could predict all sentences as negative and still have 90% accuracy. Set balancing i.e. intentionally dropping negative sentences to balance the ratio, didn’t help as it increased false positive predictions. Therefore, accuracy isn't a very useful indicator as it is always more than 90% when most sentences are predicted to be negative. Cohen's kappa is a better indicator of the results.

Lines from Biology-OP were the training set and lines from all three books were used to form three testing sets. When Biology-OP was the testing set there was a very high chance of one or both concepts from the testing pairs being in the training set. Similarly, when Concepts of Biology-OP was the testing set since the book has a similar subject matter there was a good chance of one or both concepts being in the training set. When College Physics was the testing set, it had a different subject matter and there was a low chance of either concept being in the training set. Naive Bayes was providing 60-70% Kappa for the lines from the same textbook i.e. Biology-OP, 50-60% Kappa for Concepts of Biology-OP and 30-50% kappa for College Physics. Logistic regression produced better results. With logistic regression, the results showed 80-85% kappa for same book Biology-OP and produced 75-80% kappa for both Concepts of Biology-OP and College Physics.

Chapter 4

Implementation and Testing

The books used for experimentation were OpenStax Biology-OP, Concepts of Biology-OP and College Physics. Training set solely consisted of lines extracted from Biology-OP while testing sets were made up of sentences extracted from all three books including Biology-OP. However, the sentences in Biology-OP training and testing sets were unique.

To see each stage of the process let us consider the concept ‘vaccine’ from the appendix of Biology-OP. Some other concepts from the appendix of the textbook that match with vaccine are resistance, system and virus. Following are the sentences extracted from the textbook after coreference resolution.

	key1	key2	text
1	vaccine	resistance	eventually transgenic plants may be engineered to produce vaccine antigens that can be eaten to confer disease resistance.
2	vaccine	resistance	importantly mucosal-administered vaccines elicit both mucosal and systemic immunity and produce the same level of disease resistance as injected vaccines.
3	vaccine	system	importantly mucosal-administered vaccines elicit both mucosal and systemic immunity and produce the same level of disease resistance as injected vaccines.

4	vaccine	system	viral vaccines may also be used in active viral infections boosting the ability of the immune system to control or destroy the virus.
5	vaccine	virus	vaccines for prevention while we do have limited numbers of effective antiviral drugs, such as those used to treat hiv and influenza, the primary method of controlling viral disease is by vaccination, which is intended to prevent outbreaks by building immunity to a virus or virus family
6	vaccine	virus	the danger of using live vaccines, which are usually more effective than killed vaccines, is the low but significant danger that these viruses will revert to these viruses disease-causing form by back mutations.
7	vaccine	virus	the concept behind this is that by giving the vaccine, immunity is boosted without adding more disease-causing virus.
8	vaccine	virus	many of diseases in humans can be prevented by the use of viral vaccines, which stimulate protective immunity against the virus without causing major disease.
9	vaccine	virus	unfortunately, these recognition sites on hiv change at a rapid rate because of mutations, making the production of an effective vaccine against the virus very difficult, as the virus evolves and adapts.
10	vaccine	virus	using newly developed vaccines that boost the immune response in this way, there is hope that affected individuals will be better able to control the virus, potentially saving a greater percentage of infected persons from a rapid and very painful death.
11	vaccine	virus	some vaccines are in continuous development because certain viruses, such as influenza and hiv, have a high mutation rate compared to other viruses and normal host cells.
12	vaccine	virus	live vaccines are usually made by attenuating (weakening) the “wild-type” (disease-causing) virus by growing attenuating (weakening) in the laboratory in tissues or at temperatures different from what the virus is accustomed to in the host.
13	vaccine	virus	viral vaccines may also be used in active viral infections, boosting the ability of the immune system to control or destroy the virus.

14	vaccine	virus	the killed viral vaccines and subunit viruses are the killed viral vaccines and subunit viruses incapable of causing disease.
15	vaccine	virus	a variety of diseases in animals, including humans, ranging from the common cold to potentially fatal illnesses like meningitis can be treated by antiviral drugs or by vaccines, but some viruses, such as hiv, are capable of both avoiding the immune response and mutating to become resistant to antiviral drugs.

Table 2: Sample of extracted sentences

Next step of the process is dependency fragment extraction. After parsing each sentence through spaCy's dependency parser and generating dependency trees, the shortest path between the two concepts is selected and the words are put back into their original order incase they were jumbled up during dependency tree generation and tree traversal.

	key1	key2	text	dependencyFragment
1	vaccine	resistance	eventually transgenic plants may be engineered to produce vaccine antigens that can be eaten to confer disease resistance.	engineered produce vaccine antigens eaten confer resistance
2	vaccine	resistance	importantly mucosal-administered vaccines elicit both mucosal and systemic immunity and produce the same level of disease resistance as injected vaccines.	vaccines elicit produce level of resistance
3	vaccine	system	importantly mucosal-administered vaccines elicit both mucosal and systemic immunity and produce the same level of disease resistance as injected vaccines.	vaccines elicit mucosal systemic immunity
4	vaccine	system	viral vaccines may also be used in active viral infections boosting the ability of the immune system to control or destroy the virus.	vaccines used in infections boosting ability of system
5	vaccine	virus	vaccines for prevention while we do have limited numbers of effective antiviral drugs, such as those used to treat hiv and influenza, the primary method of	credit modification of work by

			controlling viral disease is by vaccination, which is intended to prevent outbreaks by building immunity to a virus or virus family	vaccines is by vaccination intended prevent by building immunity to virus
6	vaccine	virus	the danger of using live vaccines, which are usually more effective than killed vaccines, is the low but significant danger that these viruses will revert to these viruses disease-causing form by back mutations.	danger of using vaccines is danger viruses revert
7	vaccine	virus	the concept behind this is that by giving the vaccine, immunity is boosted without adding more disease-causing virus.	is by giving vaccine boosted without adding virus
8	vaccine	virus	many of diseases in humans can be prevented by the use of viral vaccines, which stimulate protective immunity against the virus without causing major disease.	prevented by use of vaccines stimulate immunity against virus
9	vaccine	virus	unfortunately, these recognition sites on hiv change at a rapid rate because of mutations, making the production of an effective vaccine against the virus very difficult, as the virus evolves and adapts.	sites making production of vaccine against virus difficult
10	vaccine	virus	using newly developed vaccines that boost the immune response in this way, there is hope that affected individuals will be better able to control the virus, potentially saving a greater percentage of infected persons from a rapid and very painful death.	using vaccines is hope be able control virus
11	vaccine	virus	some vaccines are in continuous development because certain viruses, such as influenza and hiv, have a high mutation rate compared to other viruses and normal host cells.	vaccines are viruses have
12	vaccine	virus	live vaccines are usually made by attenuating (weakening) the “wild-type” (disease-causing) virus by growing attenuating (weakening) in the laboratory in tissues or at temperatures different from what the virus is accustomed to in the host.	vaccines made by attenuating virus
13	vaccine	virus	viral vaccines may also be used in active viral infections, boosting the ability of the immune system to control or destroy the virus.	vaccines used boosting ability control destroy virus

14	vaccine	virus	the killed viral vaccines and subunit viruses are the killed viral vaccines and subunit viruses incapable of causing disease.	vaccines viruses are
15	vaccine	virus	a variety of diseases in animals, including humans, ranging from the common cold to potentially fatal illnesses like meningitis can be treated by antiviral drugs or by vaccines, but some viruses, such as hiv, are capable of both avoiding the immune response and mutating to become resistant to antiviral drugs.	variety ranging from to treated by vaccines viruses are

Table 3: Generation of dependency fragments for the sample set

Next, we generate the fragment_dep, fragment_POS features using spaCy:

	dependencyFragment	fragment_dep	fragment_POS
1	engineered produce vaccine antigens eaten confer resistance	ROOT xcomp compound dobj relcl xcomp dobj	VERB VERB NOUN NOUN VERB VERB NOUN
2	vaccines elicit produce level of resistance	compound ROOT conj dobj prep pobj	NOUN VERB VERB NOUN ADP NOUN
3	vaccines elicit mucosal systemic immunity	compound ROOT dobj amod conj	NOUN VERB NOUN ADJ NOUN
4	vaccines used in infections boosting ability of system	nsubjpass ROOT prep pobj acl dobj prep pobj	NOUN VERB ADP NOUN VERB NOUN ADP NOUN
5	credit modification of work by vaccines is by vaccination intended prevent by building immunity to virus	nsubj appos prep pobj prep pobj ROOT prep pobj relcl xcomp prep pcomp dobj prep pobj	NOUN NOUN ADP NOUN ADP NOUN VERB ADP NOUN VERB VERB ADP VERB NOUN ADP NOUN
6	danger of using vaccines is danger viruses revert	nsubj prep pcomp dobj ROOT attr nsubj relcl	NOUN ADP VERB NOUN VERB NOUN NOUN VERB
7	is by giving vaccine boosted without adding virus	ROOT prep pcomp dobj ccomp prep pcomp dobj	VERB ADP VERB NOUN VERB ADP VERB NOUN
8	prevented by use of vaccines stimulate immunity against virus	ROOT agent pobj prep pobj relcl dobj prep pobj	VERB ADP NOUN ADP NOUN VERB NOUN ADP NOUN
9	sites making production of vaccine against virus difficult	ROOT acl nsubj prep pobj prep pobj ccomp	NOUN VERB NOUN ADP NOUN ADP NOUN ADJ
10	using vaccines is hope be able control virus	advcl dobj ROOT attr acl acomp xcomp dobj	VERB NOUN VERB NOUN VERB ADJ VERB NOUN
11	vaccines are viruses have	nsubj ROOT nsubj advcl	NOUN VERB NOUN VERB

12	vaccines made by attenuating virus	nsubjpass ROOT agent pcomp dobj	NOUN VERB ADP VERB NOUN
13	vaccines used boosting ability control destroy virus	nsubjpass ROOT advcl dobj acl conj dobj	NOUN VERB VERB NOUN VERB VERB NOUN
14	vaccines viruses are	nsubj conj ROOT	NOUN NOUN VERB
15	variety ranging from to treated by vaccines viruses are	nsubj acl prep prep pcomp agent pobj conj ROOT	NOUN VERB ADP ADP VERB ADP NOUN NOUN VERB

Table 4: Generation of fragment_dep and fragment_POS features for the sample set

Then we count the nouns, adpositions, adjectives, verbs and adverbs to generate the count_NOUN, count_ADP, count_ADJ, count_VERB and count_ADV.

	dependencyFragment	count_NOUN	count_ADP	count_ADJ	count_ VERB	count_ADV
1	engineered produce vaccine antigens eaten confer resistance	3	0	0	4	0
2	vaccines elicit produce level of resistance	3	1	0	2	0
3	vaccines elicit mucosal systemic immunity	3	0	1	1	0
4	vaccines used in infections boosting ability of system	4	2	0	2	0
5	credit modification of work by vaccines is by vaccination intended prevent by building immunity to virus	7	5	0	4	0
6	danger of using vaccines is danger viruses revert	4	1	0	3	0
7	is by giving vaccine boosted without adding virus	2	2	0	4	0
8	prevented by use of vaccines stimulate immunity against virus	4	3	0	2	0

9	sites making production of vaccine against virus difficult	4	2	1	1	0
10	using vaccines is hope be able control virus	3	0	1	4	0
11	vaccines are viruses have	2	0	0	2	0
12	vaccines made by attenuating virus	2	1	0	2	0
13	vaccines used boosting ability control destroy virus	3	0	0	4	0
14	vaccines viruses are	2	0	0	1	0
15	variety ranging from to treated by vaccines viruses are	3	3	0	3	0

Table 5:Generation of count_NOUN, count_ADP, count_ADJ, count_VERB and count_ADV features for the sample set

The <dependencyFragment, text, key1, key2, fragment_dep, fragment_POS, count_NOUN, count_ADP, count_ADJ, count_VERB, count_ADV > when combined form the testing set.

The ‘class’ field is added if we require validation i.e. accuracy and kappa calculation. In the training set this field is used to train the logistic regression model.

	class	dependencyFragment	text
1	neg	engineered produce vaccine antigens eaten confer resistance	eventually transgenic plants may be engineered to produce vaccine antigens that can be eaten to confer disease resistance.
2	neg	vaccines elicit produce level of resistance	importantly mucosal-administered vaccines elicit both mucosal and systemic immunity and produce the same level of disease resistance as injected vaccines.
3	pos	vaccines elicit mucosal systemic immunity	importantly mucosal-administered vaccines elicit both mucosal and systemic immunity and produce the same level of disease resistance as injected vaccines.

4	neg	vaccines used in infections boosting ability of system	viral vaccines may also be used in active viral infections boosting the ability of the immune system to control or destroy the virus.
5	neg	credit modification of work by vaccines is by vaccination intended prevent by building immunity to virus	vaccines for prevention while we do have limited numbers of effective antiviral drugs, such as those used to treat hiv and influenza, the primary method of controlling viral disease is by vaccination, which is intended to prevent outbreaks by building immunity to a virus or virus family
6	neg	danger of using vaccines is danger viruses revert	the danger of using live vaccines, which are usually more effective than killed vaccines, is the low but significant danger that these viruses will revert to these viruses disease-causing form by back mutations.
7	neg	is by giving vaccine boosted without adding virus	the concept behind this is that by giving the vaccine, immunity is boosted without adding more disease-causing virus.
8	neg	prevented by use of vaccines stimulate immunity against virus	many of diseases in humans can be prevented by the use of viral vaccines, which stimulate protective immunity against the virus without causing major disease.
9	neg	sites making production of vaccine against virus difficult	unfortunately, these recognition sites on hiv change at a rapid rate because of mutations, making the production of an effective vaccine against the virus very difficult, as the virus evolves and adapts.
10	neg	using vaccines is hope be able control virus	using newly developed vaccines that boost the immune response in this way, there is hope that affected individuals will be better able to control the virus, potentially saving a greater percentage of infected persons from a rapid and very painful death.
11	neg	vaccines are viruses have	some vaccines are in continuous development because certain viruses, such as influenza and hiv, have a high mutation rate compared to other viruses and normal host cells.
12	pos	vaccines made by attenuating virus	live vaccines are usually made by attenuating (weakening) the “wild-type” (disease-causing) virus by growing attenuating (weakening) in the laboratory in tissues or at temperatures different from what the virus is accustomed to in the host.

13	neg	vaccines used boosting ability control destroy virus	viral vaccines may also be used in active viral infections, boosting the ability of the immune system to control or destroy the virus.
14	neg	vaccines viruses are	the killed viral vaccines and subunit viruses are the killed viral vaccines and subunit viruses incapable of causing disease.
15	neg	variety ranging from to treated by vaccines viruses are	a variety of diseases in animals, including humans, ranging from the common cold to potentially fatal illnesses like meningitis can be treated by antiviral drugs or by vaccines, but some viruses, such as hiv, are capable of both avoiding the immune response and mutating to become resistant to antiviral drugs.

Table 6: Adding class label for training set.

For this 15-sentence test case LightSide's logistic regression algorithm gives us the following predictions as output. Here 'class' is the manually labeled and 'class_predition' is the prediction generated.

	class	class_prediction	dependencyFragment
1	neg	neg	engineered produce vaccine antigens eaten confer resistance
2	neg	neg	vaccines elicit produce level of resistance
3	pos	pos	vaccines elicit mucosal systemic immunity
4	neg	neg	vaccines used in infections boosting ability of system
5	neg	neg	credit modification of work by vaccines is by vaccination intended prevent by building immunity to virus
6	neg	neg	danger of using vaccines is danger viruses revert
7	neg	neg	is by giving vaccine boosted without adding virus
8	neg	neg	prevented by use of vaccines stimulate immunity against virus

9	neg	neg	sites making production of vaccine against virus difficult
10	neg	neg	using vaccines is hope be able control virus
11	neg	neg	vaccines are viruses have
12	pos	pos	vaccines made by attenuating virus
13	neg	pos	vaccines used boosting ability control destroy virus
14	neg	neg	vaccines viruses are
15	neg	neg	variety ranging from to treated by vaccines viruses are

Table 7: class_prediction generated by LigthSide's Logistic regression algorithm.

For example Table 7's output can be converted into a confusion matrix which shows 2 true positives or TP (both the predicted and actual label are pos), 12 true negatives or TN (both the predicted and actual labels are neg), 1 false positives or FP (the manual label is neg but the prediction is pos), and 0 false negatives or FN (the manual label is pos but the prediction is neg). We can calculate the accuracy and Cohen's Kappa as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$P_o = Accuracy$$

$$P_{Pos} = \frac{TP + FP}{TP + TN + FP + FN} \times \frac{TP + FN}{TP + TN + FP + FN}$$

$$P_{Neg} = \frac{TN + FP}{TP + TN + FP + FN} \times \frac{TN + FN}{TP + TN + FP + FN}$$

$$P_e = P_{Pos} + P_{Neg}$$

$$Kappa = \frac{P_o - P_e}{1 - P_e}$$

Using this we can calculate that the accuracy for the above test case is 93.33% and Cohen's Kappa is 76%. The output would be the three positive predictions (although one of them is a false positive). We can summarize this output as the concept map depicted in Figure 11:

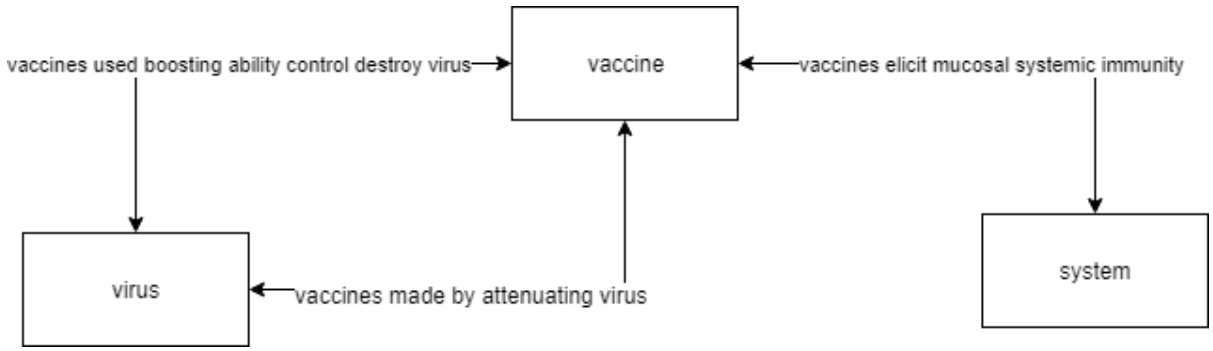

Figure 11: Concept map output.

Table 8, Table 9 and Table 10 show the results of tests performed on the three textbooks Biology-OP, Concepts of Biology-OP and College Physics. The training set was derived from the from Biology-OP textbook using 1110 unique appendix entries as concepts. Using pairs of concepts from this set we generated 4512 sentences that were used to train the logistic regression model in LightSide. These sentences were manually labeled in the 'class' field of the input CSV as 'pos' i.e. the sentence fragment correctly describes the relation between the concepts and 'neg' i.e. the sentence does not describe the relation correctly. Test sets from each book were also generated similarly. After logistic regression, LightSide stores the predicted labels in the 'class_prediction' field of the output CSV. This 'class_prediction' field can be then compared to the assigned labels in the 'class' field of test set to generate confusion matrix and calculate the accuracy and Cohen's kappa.

Result set 1: Training Set 4512 sentences from Biology-OP. Testing Set 1811 sentences from Biology-OP.

Model Evaluation Metrics		Model Confusion Matrix		
Metric	Value	Label\Predicted	Neg	Pos
Accuracy	97.29%	Neg	1570	14
Kappa	87.15%	Pos	35	192

Table 8: Biology-OP testing set of 1811 sentences

Result set 2: Training Set 4512 sentences from Biology-OP. Testing Set 1340 sentences from Concepts of Biology-OP.

Model Evaluation Metrics		Model Confusion Matrix		
Metric	Value	Label\Predicted	Neg	Pos
Accuracy	96.19%	Neg	1191	43
Kappa	77.30%	Pos	8	98

Table 9: Concepts of Biology-OP testing set of 1340 sentences

Result set 3: Training Set 4512 sentences from Biology-OP. Testing Set 1500 sentences from College Physics.

Model Evaluation Metrics		Model Confusion Matrix		
Metric	Value	Label\Predicted	Neg	Pos
Accuracy	98.00%	Neg	1399	30
Kappa	81.53%	Pos	0	71

Table 10: College Physics testing set of 1500 sentences

Chapter 5

Conclusion and Future Works

In this thesis, we have described an approach toward automatically generating a concept map from an e-textbook using the appendix as the source for concepts. Automating the key part of this approach, i.e. determining the relationships between any pair of concepts, through NLP and machine learning was described and illustrated. We have demonstrated that relations between concepts can be mined using natural language concepts of dependency trees and parts of speech tagging along with logistic regression algorithm for machine learning. This auto-generated concept can save time by providing a starting point to the person generating the concept map.

This system, however, needs further refinement as it produces false positives and false negatives in 30 percent of the sentences, thus sometimes missing out on relations or showing ill-defined relations. We hope that with a larger and more varied training set the system's accuracy will increase. In the future, we would also like the system to be able to generate concept maps with varying levels of complexities as opposed to the single detailed concept map the current system generates. We would also like to add a concept mining module to the system for textbooks that do not contain an appendix.

The current prototype can serve as a basis for further development in this field of research of concept map mining and semi-automatic generation of concept map based textbooks.

Bibliography

[1] P. J. Hager and . N. C. Corbin, Designing & Delivering: Scientific, Technical, and Managerial Presentations, 1997.

[2] J. A. Turns, C. Atman and R. S. Adams, "Concept maps for engineering education: a cognitively motivated tool supporting varied assessment functions," *IEEE Transactions on Education*, vol. 43, no. 2, pp. 164-173, May 2000.

[3] J. D. Novak and A. . J. Cañas, "The Theory Underlying Concept Maps and How to Construct and Use Them.," 2008.

[4] D. Meyers, "Concept Mapping in Chemistry," chemedx, 15 Jun 2016 . [Online]. Available: https://www.chemedx.org/blog/concept-mapping-chemistry. [Accessed june 2019].

[5] K. Žubrinic´, University of Dubrovnik, Department of electrical engineering and computing, [Online]. Available: http://bib.irb.hr/datoteka/578611.KZubrinic-KvalifikacijskiRad.pdf. [Accessed June 2018].

[6] R. B. Clariana and R. Koul, "A computer-based approach for translating text into concept map-like representations.," in *Proceedings of the First International Conference on Concept Mapping*, Pamplona, Spain, 2004.

[7] R. Richardson and E. Fox, "Using concept maps in digital libraries as a cross-language resource discovery tool," in *Digital Libraries, Joint Conference on(JCDL)*, Denver, CO, USA, 2007.

[8] D. Crystal, A dictionary of linguistics and phonetics 4th edition, Cambridge, MA: Blackwell Publishing, 1997.

[9] Stanford NLP Group, "Neural Network Dependency Parser," November 2018. [Online]. Available: https://nlp.stanford.edu/software/nndep.html.

[10] J. V. Jorge and A. C. Rafael, "Concept Map Mining: A definition and a framework for its evaluation," in *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, 2008.

[11] K. Žubrinic´, "Automatic creation of a concept map," University of Dubrovnik, Department of electrical engineering and computing, Dubrovnik, Croatia.

[12] Project Gutenberg, "Project Gutenberg," 2019. [Online]. Available: https://www.gutenberg.org. [Accessed 2019].

[13] OpenStax, "OpenStax," 2019. [Online]. Available: https://openstax.org/subjects/science. [Accessed 2019].

[14] The Stanford Natural Language Processing Group, "Stanford Deterministic Coreference Resolution System," 7 February 2016. [Online]. Available: https://nlp.stanford.edu/software/dcoref.shtml. [Accessed 1 June 2019].

[15] S. Martschat and M. Strube, "Cort A toolkit for coreference resolution and error analysis.," smartschat, 4 November 2015. [Online]. Available: https://github.com/smartschat/cort. [Accessed 1 June 2019].

[16] T. Wolf, "Fast Coreference Resolution in spaCy with Neural Network," Huggingface Inc., 18 April 2019. [Online]. Available: https://github.com/huggingface/neuralcoref. [Accessed 1 June 2019].

[17] Google, "Textsum," January 2018. [Online]. Available: https://github.com/tensorflow/models/tree/master/research/textsum. [Accessed June 2019].

[18] N. Rotem, "Open Text Summarizer," 2003. [Online]. Available: https://github.com/neopunisher/Open-Text-Summarizer. [Accessed 2019].

[19] Resoomer, "Resoomer | Summarizer to make an automatic text summary online," Resoomer, 2019. [Online]. Available: https://resoomer.com/en/) . [Accessed June 2019].

[20] Smmry, "Smmry," 2019. [Online]. Available: https://smmry.com/). [Accessed June 2019].

[21] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E. D. Trippe, J. B. Gutierrez and K. Kochut, "Text Summarization Techniques: A Brief Survey," *arXiv*, July 2017.

[22] M. Hassel, *Resource Lean and Portable Automatic Text Summarization,* Stockholm, Sweden: KTH School of Computer Science and Communication, 2007.

[23] D. Gaikwad, "A Review Paper on Text Summarization," *International Journal of Advanced Research in Computer and Communication Engineering,* vol. 5, no. 3, March 2016.

[24] Y. J. Kumar, O. S. Goh, H. Basiron, N. H. Choon and P. C. Suppiah, "A Review on Automatic Text Summarization Approaches," *Journal of Computer Science,* vol. 12, no. 4, pp. 178-190, 2016.

[25] M. A. Fattah and F. Ren, "Automatic Text Summarization," *International Journal of Computer and Information Engineering,* vol. 2, no. 1, pp. 90-93, 2008.

[26] Splunk, "SIEM, AIOps, Application Management, Log Management, Machine Learning, and Compliance | Splunk," 2019. [Online]. Available: https://www.splunk.com/. [Accessed 2019].

[27] Scikit-learn, "scikit-learn: machine learning in Python — scikit-learn 0.21.2 documentation," May 2019. [Online]. Available: https://scikit-learn.org/stable/index.html. [Accessed June 2019].

[28] Weka, " Weka 3 - Data Mining with Open Source Machine Learning Software in Java," 2018. [Online]. Available: https://www.cs.waikato.ac.nz/ml/weka/index.html. [Accessed 2019].

[29] Keras, "Home - Keras Documentation," 2019. [Online]. Available: https://keras.io/. [Accessed 2019].

[30] Google, "Tensorflow," 2019. [Online]. Available: https://www.tensorflow.org. [Accessed 2019].

[31] M. Kang, S. Chaudhuri, Y.-C. Wang, M. Joshi, E. Rosé, M. Van Velsen and C. P. Rosé, "LightSide Researcher's Workbench," 2019. [Online]. Available: http://ankara.lti.cs.cmu.edu/side/. [Accessed 2019].

[32] NLTK Project, "Natural Language Toolkit — NLTK 3.4.3 documentation," 2019. [Online]. Available: https://www.nltk.org/. [Accessed 2019].

[33] TextRazor, "TextRazor - The Natural Language Processing API," 2019. [Online]. Available: https://www.textrazor.com. [Accessed 2019].

[34] spaCy, "spaCy · Industrial-strength Natural Language Processing in Python," 2019. [Online]. Available: https://spacy.io/. [Accessed 2019].