APPLYING LEXICAL SUBSTITUTION AND TEXT MINING FOR BIOINSPIRED ENGINEERING DESIGN

A Dissertation

by

SOOYEON LEE

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Daniel A. McAdams
Committee Members,	James Caverlee
	Richard Malak
	Michael Pate
Head of Department,	Andreas A. Polycarpou

August 2015

Major Subject: Mechanical Engineering

ABSTRACT

Nature has repeated its evolution processes and developed its own "engineering" principles over a long period of time. Bioinspired design starts from the belief that nature has the most effective and optimized problem-solving schemes, which can be applied to human problems directly or indirectly. In summary, bioinspired design is the study of the design process, adapting the structure, behavior, or organic mechanisms of biology to engineering problems.

In bioinspired design studies, researchers have sought a way of improving concept generation through texts. Generally, there are two problems in text-based bioinspired design. First, there is a great lexical gap between two areas—biology and engineering. Thus, understanding the context of biological text is compromised, prohibiting analogical transfer between the two domains. Second, the amount of text is too great to be assimilated by engineers. This knowledge gap makes the engineer confused by the extensive information and slows down the design process.

The present work tried to apply lexical substitution and text mining theories to effectively process biological text. Regarding the matter of the lexical gap, this research developed an algorithm that translates biological terminology to words or phrases that are understandable to engineers by adapting four lexical sources: WordNet, Wikipedia, the Integrated Taxonomy Information System (ITIS), and WordNik. For the second problem, this research tried to categorize biological text based on morphological solutions by adapting the Latent Semantic Analysis (LSA) technique.

Two main contributions are made in this dissertation. First of all, this work is the first attempt to directly bridge the lexical gap between biology and engineering by translating biological terminology. The existing approach to bioinspired design study involves building a thesaurus or database that connects a few engineering keywords and their biological correspondences. However, since most other biological terms remain unchanged, this research is meaningful as it attempts to overcome this limitation. The second contribution is that this research ameliorates the natural language-based bioinspired concept generation. Specifically, the accessibility to biotexts for bioinspired design seems to be improved by enabling engineers to selectively acquire biological information for their problems.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my committee members, Dr. Caverlee, Dr. Malak, and Dr. Pate for their support and services for me, and Dr. Johnson who saved several times me as a substitute member. Especially, I would give my deepest thanks and appreciation to my committee chair, Dr. McAdams, for his effort and support he has dedicated to me.

To my special lab mates, Elissa Morris, Madison Burns, Shraddha S. Sawant, Wei Li, and Joanna Tsenn, I owe you so many things. Thank you so much for making my years in Texas A&M special.

I cannot say anything but thank you to my sister, Minjeong, and other friends who supported me throughout the darkest time of my life. I cannot name you all, but I am deeply indebted to your encouragement.

Finally, I give my special gratitude to my mother, Myung-yi Shin, for her love and support.

NOMENCLATURE

BNC	British National Corpus
ITIS	Integrated Taxonomy Information System
LSA	Latent Semantic Analysis
NLP	Natural Language Processing
WSD	Word Sense Disambiguation

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER I

INTRODUCTION

Humans can learn from biology. This main premise of bioinspired design, or biomimetics, has gained significant attention since the ‘60s and gained new momentum due to Janine Benyus and her book, *Biomimicry: Innovation Inspired by Nature* (Benyus, 1997). Bioinspired design is an attempt to import biological solutions, such as materials, behaviors, and structures, from nature and apply them to the engineering field to improve designs. Even though its establishment as an academic discipline has started only recently, bioinspired design has a long history, from stone blades that imitate the sharp teeth of animals to the Wright Brothers’ flying machine, which imitated the wing structure of a bird.

Relatively recent examples of bioinspired design include Velcro, gecko tape, the Mercedes-Benz Bionic concept car, or the Japanese bullet train. These engineering designs adapt elements from a cocklebur, a gecko foot, the boxfish, and the kingfisher bird, respectively, to improve efficiency and function. In addition, it has been found that biology can be a good source of inspiration for engineers in previous studies (Bonnardel, 2000; Cheong et al, 2011; Shu et al, 2011).

One benefit of incorporating biology into engineering design is that it facilitates the direct use of optimized results that have evolved over long periods of time. Biology has strategies that evolve over hundreds or even thousands years that enable organisms to adapt to their environments. Since design failure in biology can result in the extinction

of a species, every organism has its own intelligent means of survival. Bioinspired design aims to adapt these novel strategies only observed in biology to engineering design and seek ways of improving current engineering designs.

Additionally, bioinspired design enables cross-domain analogical transfer between engineering and biology. According to previous studies (Benami & Jin, 2002; Bonnardel, 2000; Qian & Gero, 1992), analogical transfer across different domains can generate creativity during the design process. Due to its natural characteristic, bioinspired design maps biological solution domains to engineering problem domains, thus facilitating analogical transfer between different domains. This indicates that bioinspired design can effectively inspire creativity in engineers during the design process.

Current bioinspired design studies

While biological systems provide a wealth of elegant and ingenious approaches to problem solving, there are several challenges that prevent designers from taking full advantage of the biological knowledge domain. One main concern is that current bioinspired design products are mostly based on ad hoc findings. This is closely related to the fact that most engineers do not have sufficient knowledge of biology to find and identify natural analogies for a given problem. Several approaches have been developed and studied to allow engineers to find and leverage biological knowledge. The efforts that have tried to establish rigorous and systematic methodologies for the bioinspired design process include BioTRIZ, keyword search, and IDEA-INSPIRE. BioTRIZ, which

was developed based on the Russian Theory of Inventive Problem Solving (TRIZ) (Al′tshuller, 1999), provides design principles for an engineering problem using two conflict design requirements. Unlike BioTRIZ, which can derive design principles directly from a problem, some approaches build databases that can deliver biological inspiration to engineers. IDEA-INSPIRE is a representative searchable database that contains natural systems and artifacts (Chakrabarti et al, 2005). It delivers biological systems based on an engineering problem that is represented using a causal behavioral model called SAPPhIRE (Srinivasan & Chakrabarti, 2009). SAPPhIRE uses a combination of verbs, nouns, and adjectives to represent behaviors.

Chakrabarti et al., Shu et al., and Nagel et al. tried to find the relation between engineering functional keywords and biologically meaningful keywords (Chakrabarti et al, 2005; Cheong et al, 2011; Nagel et al, 2010). Based on these efforts, an engineer can translate their desired engineering function into an equivalent biological function, and then search for a biological system using the biological function. Using this function-based analogy, biological systems can be mined for strategies, principles, and morphologies that can be used to solve engineering problems. The research refers to these bioinspired engineering design approaches as biological keyword searches.

Problems of current keyword search

The text corpus used in biosearching is based on biological knowledge stored in biology books or scholarly journal articles. These sources are generally written using a significant amount of technical terminology, or jargon, that is unfamiliar to engineers.

Thus, a significant obstacle to bioinspired engineering design is the difficulty engineers have in understanding descriptions of biological phenomena sufficiently to adapt them to engineering problems (Cheong et al, 2008). For example, consider a sentence offering potential biological inspiration that includes the word "Chlamydomonas." This term generally would not suggest much useful information to engineers. However, if "Chlamydomonas" is translated into "green algae," the meaning of the textual passage is more clearly conveyed to the engineer and more likely to inspire the development of a solution to a design problem.

In addition to the terminology problem, another inconvenience exists in current keyword search in that it does not highlight specific biological solutions in a massive amount of biological text. Engineering designers have to invest a huge amount of time in reading all the passages to find a biological solution, or they have to read similar information repeatedly shown in the text. These problems lower the efficiency of the design process, and as a result, they should be addressed by further studies.

Objectives and research approaches

The work presented here builds on and extends function-based biological keyword search methods. The objective of this research is to improve the concept generation phase of bioinspired design centered around function-based keyword searching (keyword searching). Based on two main concerns that exist in current keyword searches, the following two sub-objectives have been established:

1) Reducing the lexical gap between engineering and biology

- Translating biological terminologies in biotexts used in keyword searches

2) Reducing and categorizing biological information

- Finding morphological information in biotexts and cluster texts based on the explored morphology

The objective of this study to reduce the lexical gap involves treating the "jargon" of biologists and similar domains of natural science and the "jargon" of engineers like different languages. The study then creates fundamental extensions of current lexical substitution theory to translate biological information to information accessible to engineers. Theories of lexical substitution are used to identify the candidate words for translation (target words) and perform word sense disambiguation (WSD). In most cases, WordNet is used for WSD. However, WordNet does not contain all the biological jargon that needs to be translated. Thus, we also develop new processes to identify and translate target words not contained in WordNet by linking to Wikipedia and the Integrated Taxonomical Information System (ITIS) corpus.

Based on the second objective, the second research approach to control the amount of information returned by keyword search involves finding morphological information in biological texts and clustering those texts based on morphology. Since keyword searches are based on the functional connection between engineering and biology, finding morphological expressions in keyword search results bridges an engineering problem to morphological solutions in biology. After the morphological solutions are mined, the categorization of information becomes possible; therefore,

selecting information becomes possible. This study applies text mining theories to find biological morphologies, for which the WordNet noun categorizing scheme is used. Latent Semantic Analysis (LSA), which is widely used in information retrieval, will then be applied to biotexts to categorize them according to their morphological solutions.

CHAPTER II

BACKGROUND

This chapter introduces bioinspired design and related theories to construct an understanding of the background of the study.

Function and structure in design process

It is broadly accepted that analyzing the functions of a system in a design process is important, because function is the purpose of a system. By analyzing and decomposing systems into functions and subfunctions, designers can understand the requirements in design problems efficiently (Dieter, 1991; Miles, 1961; Nagel & Stone, 2011; Otto & Wood, 2001; Pahl et al, 2007; Ullman, 2009). Consequently, current bioinspired research for concept generation, especially biological keyword searching, searches for biological solutions by biological function, which corresponds to an engineering function.

Keyword searching enables engineers to find possible biological solutions that conduct the same purpose as an engineering system by bridging an engineering functional term to a biological functional term. However, this does not mean that biological solutions always appear in a functional form. Sometimes, the structural or behavioral form of a solution fulfills the purpose, or function, of a system physically and practically. Therefore, discovering the behavioral or structural (or morphological)

expressions buried in a text is a process of discovering biological solutions directly from biotexts.

Several different definitions of "morphology" exist according to different fields of studies. Oxford Dictionary defines morphology as "the study of the forms of things" and "a particular form, shape, or structure." As in the dictionary definition, the term "morphology" is a broad concept that encompasses the shape and structure of a system. Based on this, the research will consider physical structure (or component) and shape as morphological features. For example, expressions such as "hair" or "cone-shaped" are both considered morphological features in this study. However, chemical or biological substances or components, such as ethylene or an artery, are not considered in this research, because this research focuses on adopting analogical problem-solving schemes from biology, not just using biological substances in engineering problems.

In terms of design process research, there have been several attempts to use morphological searches in the concept generation phase (Arnold et al, 2008; Bohm et al, 2005; Bryant et al, 2005; Zwicky, 1969), yet importing biological morphology to the design space still requires further research. The benefit of importing biological morphology is that it can enhance the creativity of designs. To understand the effect of importing new morphologies to a design process, we can examine the effect of importing new structural components to a design process, bearing in mind that morphology is a broad concept that covers the structure of a system. A representative study was performed by Gero and his colleagues, who pointed out that enlarging the structural design space can be considered creative design (Gero & Kazakov, 1999). Although the

structure does not have a function and vice versa, if a designer learns the relation between function and structure, it is very hard to discard that studied knowledge (Qian & Gero, 1992). As a result, if a designer can introduce a totally new structure component to a design process, this can enlarge a design space; consequently, this new introduction can be considered creative design. Likewise, importing a biological structure, broadly morphology, to engineering design, is expected to help engineers in their creative design, since biology has unique morphologies not present in conventional engineering designs.

Functional basis

Various functional models have been developed by artificial intelligence and design theory research groups, as reviewed by Erden et al. in their paper (Erden et al, 2008). Specifically, this research uses the Functional Basis engineering function lexicon and several engineering-to-biology functional thesauri that have been developed and are under development (Chakrabarti et al, 2005; Cheong et al, 2011; Hirtz et al, 2002; Nagel & Stone, 2010; Sarkar et al, 2008).

The functional basis includes two sets of lexicons that define the engineering functions and flows required to represent engineering systems rigorously. Engineering functions and flows are represented as verbs and nouns, respectively, and flows consist of three types: Material, energy, and signal. The function refers to the transformation of flows, and it can be represented using the black box model, as shown in the Figure 1.

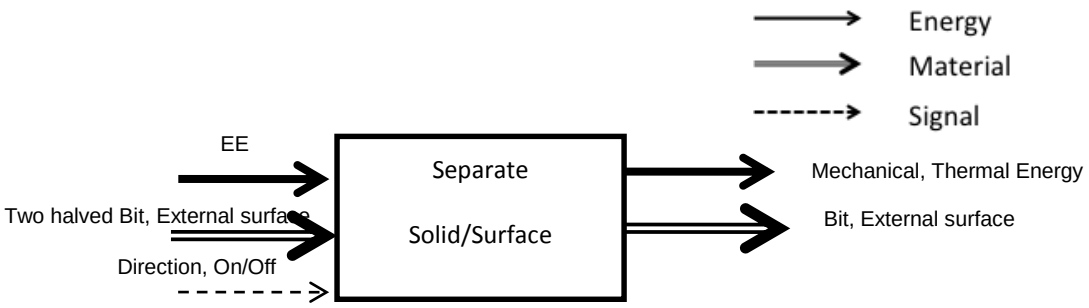


Figure 1 Black box model of an electrical drill

Using functional basis lexicons, concrete engineering problems and materials can be expressed rigorously, which facilitates engineers' concrete thinking and communications.

Figure 1 is a black box model of an electrical drill represented by the functional basis. It contains the overall purpose of the system inside the box using functional verbs and flow nouns, and the flows that pass through the system are expressed using flow nouns with arrows around the box.

Based on this black box model, engineers can analyze the system and deconstruct it into its subsystems. Figure 2 illustrates the functional model of an electrical drill deconstructed by the previous black box model to a specific model. The advantage of the functional model using the functional basis in bioinspired design is described in the study of Nagel et al. (Nagel & Stone, 2010). Their main point is that a biological system is usually in the form of abstract information; however, functional modeling can map

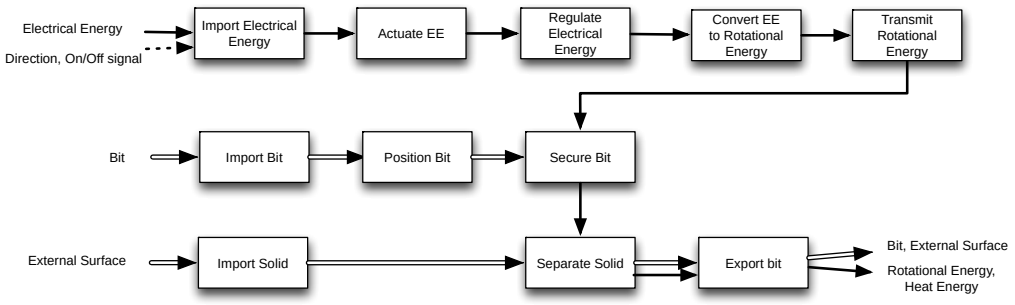


Figure 2 Functional model of an electrical drill using functional basis. This function model decomposed the system shown in the Figure 1 to its subsystem.

this abstract problem to a detailed actual form. Through this actualization, engineers are led to "created leaps" using methods such as metaphor or analogical transfer, since engineering problems and biological systems can be connected.

Current bioinspired design studies and keyword searches

Researchers have sought various ways to help engineers adopt biological principles in their work. One of the main approaches involves building a database of natural solutions that organizes these solutions according to certain schemes that allow engineers to find solutions to their engineering problems based on these schemes. Chakrabarti and his colleagues developed a knowledge-based system for bioinspired design. They focused on building a searchable database of natural solutions called IDEA-INSPIRE (Sarkar et al, 2008; Srinivasan & Chakrabarti, 2009), which categorizes engineering and biological solutions using verbs, adjectives, and nouns. Furthermore,

Vattam et al. developed a knowledge database of biological solutions called DANE (Vattam et al, 2011). This multimedia database uses functions as an index to categorize the behaviors and structures of a biological system, and it uses nodes and edges to represent super/sub-functional relationships.

The Biomimicry Institute (http://biomimicry.org) and AskNature.org (www.asknature.org) also provide biological solution databases for engineers based primarily on existing bioinspired designs. The AskNature.org knowledge base organizes existing bioinspired designs and novel natural solutions based on a taxonomy of various solutions and strategies. Some elements of the taxonomy are functional in nature; others are not.

The research presented in this study is related to finding biological analogies using textual inspiration. Text-based search methods have been developed by researchers, such as Nagel et al. (Nagel & Stone, 2010), Shu et al. (Cheong et al, 2008; Vakili & Shu, 2001), and Stroble et al. (Stroble et al, 2009). These approaches identify and return text passages that contain desired keywords. The returned text passage contains a description of the system that may provide inspiration for an engineering solution. Searching for a relevant biological text to determine if it corresponds to an engineering problem requires a linguistic connection between engineering and biological terminologies. Shu et al. established the methodology of finding natural analogies relevant to engineering functional keywords by searching biological corpora. Stroble et al. developed Biosearch using a formal functional modeling framework and a verb–noun function flow. Biosearch allows the user to search for specific engineering functions

while specifying the size of the text passage returned and other basic user features. Furthermore, Nagel and her colleagues developed the Biotext search engine, which is based on an engineering-to-biology thesaurus (Design Engineering Lab; Nagel et al, 2010).

Our approach builds on function-based keyword searching. Because of its proven usefulness, functional modeling is used in design studies as an essential procedure to achieve better design (Stone & Wood, 2000). Various functional models have been developed by artificial intelligence and design theory research groups, as reviewed by Erden et al. in their paper. Specifically, here we use the Functional Basis engineering function lexicon and several engineering-to-biology functional thesauri that have been developed and are under development (Chakrabarti et al, 2005; Cheong et al, 2011; Hirtz et al, 2002; Nagel & Stone, 2010; Sarkar et al, 2008).

CHAPTER III

LEXICAL SUBSTITUTION OF BIOLOGICAL TERMINOLOGY USING WORDNET

Motivated by the lexical gap problem in current keyword searching, the research in this chapter develops an algorithm that translates biological jargon into terms engineers will understand.

The research builds on a computational process called lexical substitution. Lexical substitution is a computational process that replaces a word with a similar word while considering the context in which the original term is used. For instance, in the sentence, "The guy who lives next door is very *taciturn*," lexical substitution can change the word *taciturn* to *laconic* or *uncommunicative*. Using lexical substitution, biological terminology can be replaced with commonly used words. This research adapts general lexical substitution theories to the bioinspired design domain.

The primary task involved in lexical substitution is finding proper candidates for an original word or a target word. In this study, the widely used English lexical thesaurus WordNet is selected as the main lexical source for collecting candidate words. However, since the target passage is highly domain-specific in this problem, it is hard to guarantee that WordNet is a valid lexical source for this task. For this reason, this chapter also determines if WordNet is adequate for biological terminology lexical substitution. For this, the coverage and ratio of vocabularies in biological texts will be examined.

Another important latent subtask in lexical substitution is finding a "correct" target word. Since many words have multiple meanings, lexical substitution should be able to determine which meanings of the original and substituted word are appropriate. Thus, lexical substitution is often closely related to the computational process of Word Sense Disambiguation (WSD). In this research, to disambiguate the meaning of a word in context, a vector space model and clustering-based WSD are used. The details of all the procedures of the lexical substitution algorithm will be described in this chapter.

Background

Background information related to the lexical substitution task for bioinspired design is described in this section.

Linguistics background

Unlike Machine Translation, which translates an entire document from one language to another, lexical substitution only targets a subset of words. Lexical substitution frequently only targets specific, known words for substitution. Here, the substitution algorithm has to start by identifying which words need to be translated. Since this research is extending biological keyword search methods, our target words are biological terminologies. The knowledge source related to this task is reviewed in this work.

British national corpus and vocabulary level

Before translating jargon, the algorithm needs to identify words that do not need to be substituted and target words for substitution. Words that do not need to be substituted are those, reasonably, in a college graduate's vocabulary. In linguistics, the term "frequency" is the frequency of occurrence of certain words in a given corpus or usage context. A frequency order for English words is available in the library of the RANGE program—a vocabulary frequency counting program developed by Paul Nation and his colleagues (Heatley et al, 1994). RANGE's corpus includes the British National Corpus (BNC), the General Service List (GSL), and the Academic Word List (AWL). Developed by Oxford University, the BNC is a well-known English corpus that contains 100 million words. Based on this, the RANGE program contains 14,999 lemmatized English words in its corpus. The GSL includes the 2,000 most common words in Standard Written English (West, 1953). The AWL includes the 570 most common words used in academic writing (Coxhead, 2000).

Chujo et al. (Chujo & Utiyama, 2006) studied the relationship between the vocabulary size of students of various grades in the United States whose first language was English and BNC frequency level. Chujo's study found that a college-level (13^{th} grade) student knows almost 60% of the top 500 words excerpted from BNC's high-frequency word list (BNC HFWL) using mutual information (MI) and McNemar's tests. These words are covered by the BNC 7000 frequency band, which means that the average 13^{th}-grade student in the United States knows 7000–8000 of the most frequently used words listed in the BNC corpus. Here, if a word is in the BNC 7000 band in the

RANGE list, the algorithm ignores it, and the rest of the words become target words for translation.

Computational background

The main substitution algorithm of this research builds on natural language processing (NLP) and word sense disambiguation (WSD). This subsection introduces computational background information related to these theories.

Natural language processing

Natural language (NL) is the language naturally developed and spoken by humans. NLP studies the theory and principles of natural languages and converts them to a form that artificial intelligence can process. NLP is applied not only in computer science, but also in linguistics and bioinformatics. For example, a study of medical informatics used NLP to improve understanding of a medical corpus (Spyns, 1996), and a biomedical study used NLP to classify and retrieve information in biological corpus texts or literature (Krallinger et al, 2005). In bioinspired design research, Hacco and Shu applied NLP to a design study to extract non-technical keywords for keyword searching (Hacco & Shu, 2002), and Cheong et al. used NLP to extract biologically meaningful keywords (Cheong et al, 2008). In the present research, NLP techniques, such as a sentence parsing, are used in the algorithm to preprocess the biological text for lexical substitution.

WordNet

The basic lexical corpus used in this research is WordNet. WordNet is an English lexicon database developed at Princeton University (Fellbaum, 2010), and is widely used in the Computer Science and Linguistic fields. Shu et al. used WordNet to identify the biologically meaningful keywords in their research on bioinspired design (Cheong et al, 2008). WordNet's strength as a lexical source for NLP is due to its classification of relations between relevant words. The basic relation set in WordNet is a synset. A synset is a group of words with similar meanings. A synset is related to other synsets through conceptual and semantic relations. The two representative relations are the superordinate–subordinate relations. Conceptually, the superordinate term, also known as a hypernym, encompasses the subordinate term, known as a hyponym. This super-subordinate relation is also called an ISA relation (Fellbaum, 2010). For example, 'water' is a hypernym of 'tap water' and 'tap water' is a hyponym of 'water'. In this research, a combination of synset and hypernym sets will be used in the translation process.

Lexical substitution

Lexical substitution aims to replace a word (target word) with a contextually appropriate alternative word (McCarthy & Navigli, 2007; 2009). Lexical substitution is applied and benefits various NLP areas, such as lexical acquisition, information retrieval, or machine translation (Hassan et al, 2007). Lexical substitution mostly consists of two main tasks:

1) Finding candidate substitutions, and

2) Selecting the best candidate among the substitutions considering the context (McCarthy & Navigli, 2009).

The need to find candidate substitutions indicates that lexical sources need to be found for this research. The lexical source could be a dictionary, a web-based source, or a combination of both. This research uses WordNet as an initial lexical source for finding candidate words. If a target word has only one meaning, a simple substitution can be performed. If the target word has multiple meanings, the lexical substitution task must consider the context of the target word to determine the correct usage of it. Thus, task two is closely related to WSD.

WSD is a field in NLP that identifies the correct contextual meaning of a word. Humans solve polysemy problems by considering the words adjacent (in text or speech) to the one under consideration. Humans recognize the word ‘apple’ as a fruit if words like ‘eat’ and ‘red’ are located in a sentence with a word ‘apple’. In contrast, humans recognize ‘apple’ as a company name if the word ‘phone’, ‘computer’, or ‘system’ are in the same sentence. Similarly, the WSD process tries to determine the sense of a word in a context using a computational or statistical method. In the present research, K-means clustering based WSD is used for its computational efficiency (Schutze, 1998). Details of the WSD algorithm will be explain later in the Algorithm subchapter.

Lexical substitution systems that have been proposed based on WordNet

This section offers a review of other lexical substitution models that use WordNet here. Most of the systems described in this chapter are from SEMEVAL-2007 English Lexical Substitution Task (McCarthy and Navigli 2007). The approaches and methodologies of each system are described briefly below.

Giuliano et al. (Giuliano et al, 2007) proposed two lexical substitution systems, IRST1-lsa and IRST2-syn. These systems used WordNet 2.0 and the Oxford American Writer Thesaurus (first edition) as sources of candidate words. IRST1 ranked candidate words using LSA, and IRST2 used Google web 1T 5-gram frequency. Yuret proposed another lexical substitution and WSD system, called KU, in SENSEVAL-2007 (Yuret, 2007). KU uses Roget's Thesaurus and WordNet to collect candidate words, and ranks them based on a statistical language model using web 1T 5-gram frequency. Another lexical substitution system, UNT, collects candidate words from WordNet (synsets and hypernyms), Encarta, Roget's, and a bi-lingual dictionary (Hassan et al, 2007). Its algorithm is based on back-and-forth translation between the dictionaries of each language. In addition, UNT introduced an n-gram based model using Google web IT corpus.

HIT is a lexical substitution system proposed by Zhao et al. in SENSEVAL-2007, collects candidate words from WordNet's synonyms and hypernyms, and ranks those collected candidate words based on google queries (Zhao et al, 2007). Martinez and his colleagues proposed the system named MELB (Martinez et al, 2007). MELB's candidate words are synonyms, immediate hypernym, and two-step hypernyms in

WordNet. MELB uses its unique Semcore-based filter, and this filter removes the candidates that are related to infrequent sense in Semcore. Ranking strategy is based on Google query counts.

Sinha and Mihalcea reviewed and experimented with the above mentioned systems, and developed their own lexical substitution system (Sinha & Mihalcea, 2014). They examined various lexical sources for lexical substitution problems, including Roget's Thesaurus, WordNet, TransGraph, Lin's distributional similarity, and Encarta. They applied LSA after generating a candidate-context matrix including the target word itself as a part of the context, thereby calculating similarity between a candidate word and a target word. They also used Explicit Semantic Analysis (ESA) (Gabrilovich & Markovitch, 2007) with a Term Frequency - Inverse Document Frequency (TF-IDF) score and calculated candidate-target similarity as in the LSA. TF-IDF is one of the most widely used weighting schemes in information retrieval, which reflects importance of a word in a corpus or a document. In addition, Google N-gram model was examined with various n-grams include 3-gram, 4-gram, and so on. They concluded that Encarta and WordNet provide best performance among lexical sources and decision lists method is the best in supervised methods.

Word Sense Disambiguation with clustering

Word-sense disambiguation (WSD) is a computational process that identifies the correct contextual meaning of a word. WSD plays an important role in lexical substitution tasks, because many words have multiple meanings (i.e., they are

polysemic), and their meaning in context should be identified before the substitute task. In this research, K-means clustering based WSD is used for its efficiency (Schutze, 1998).

K-means clustering

Details of K-mean clustering are explained in Figure 3. Assume that representative vector is $\vec{x}$. The goal of a K-mean algorithm is assigning all data into one of k clusters by finding a set of S_k that minimize Residual Sum of Squares (RSS) (Manning et al, 2008). RSS can be expressed as

$RSS = \sum_{i=1}^{k} \sum_{\vec{x} \in S_k} \|\vec{x} - \mu_i\|^2$,

where μ_i and S_k represent a mean of i-th cluster and the set of k-th cluster, respectively. Initially, means are arbitrarily assigned, then the program updates a centroid of each cluster by the following equation:

$\vec{\mu} = \frac{1}{|s|} \sum_{\vec{x} \in s_k} \vec{x}$

The clustering iteratively refines the mean and the cluster set S_k, until a stop criterion is met. A common stop criterion of K-mean clustering is a convergence of an RSS value, and this is used in the present case.

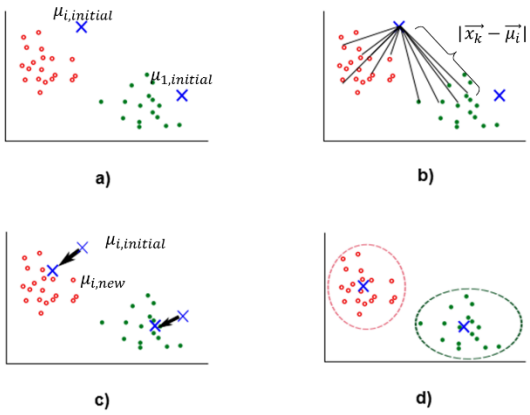


Figure 3 Outline of K-mean Clustering: a) The program initially guesses a mean of each cluster, and means are marked as a cross in the figure. The number of cluster is predetermined. b) First, the program compares the distance between each data and the initial mean, and assigns data to the closest cluster. c) At step c, relocation of the mean of a cluster is performed. d) Repeat b and c step iteratively until the stop criterion holds (Manning et al, 2008)

Problem clarification and research approach

In the context of engineering design, this research is rooted on the concept generation phase. Additionally, the study builds on a function-based approach in which the design problem is abstracted in terms of the functions the engineered system is to provide. In this context, the concept generation effort is focused on finding principles, forms, and mechanisms that provide the needed function.

Given this function-based approach, bioinspired design is a valuable area. The premise is that nature can be searched for principles, forms, and mechanisms that provide the required function in a given design problem.

Based on functional analogy, researchers – including the authors of this research and others – have made significant progress towards guiding engineers to relevant and analogous biological systems. In short, progress has been made in creating a thesaurus to bridge the lexicon of engineering functions and the terminology of functions in biology and natural science. Engineering functions are used precisely as they are defined in the Functional Basis (Hirtz et al, 2002). The current engineering function to biological function translation is the result of multiple parallel efforts (Cheong et al, 2011; Nagel et al, 2010; Srinivasan & Chakrabarti, 2009). Building on this thesaurus, a search algorithm has been created by researchers that allows an engineer to search through text-based knowledge sources such as books and journals to find natural functions (Nagel & Stone, 2011). Using these results, engineers are able to find a natural function that is equivalent, or analogous, to the needed engineering function. Thence, engineers are able to develop possible adaptations to reach an engineered solution.

What the current state of the art does not do is identify what that system is or describe the causal mechanism in an engineering lexicon. For example, an engineer may be seeking a solution to a problem with the function of "regulate fluid." In this case, the function is "regulate" and (one of) the biological keywords is "respire." Using the search tool, the engineer enters "regulate" and the search engine returns text passages that contain the term respire. The engineer then reads the returned passages to find the agent that performs the function "respire" and then must understand the specific mechanism that the agent employs to "respire."

The description of these natural systems is written using the technical language of biologists, entomologists, or similar natural scientists. Though engineers have a deep knowledge of physics, physical principles, and solution archetypes, we have found that the description of the natural system's solution principle is still typically difficult for an engineer to understand; biologists and engineers use language that differs in more than the lexicon of function.

Removal of this language disconnection between biologists and engineers brings us to the focus of the present research. We need to create a Lexical Substitution system that translates biological and similar corpora into the lexicon of engineering design. Combined with the keyword search algorithms, a substitution engine can present engineers with understandable descriptions of natural systems likely to suggest a solution to an engineering design problem. As will be outlined in our approach below, this language substitution problem creates a set of specific subproblems, the solutions of which are presented in this research.

Table 3 shows the result generated from a search using the engineering function search term "export" (Glier et al, 2013). The biological functional equivalents are "bind", "block", "breakdown", "excrete", and "inactivate" (Cheong et al, 2011). In this table, words commonly used in biology that are not typically familiar to engineers are highlighted. Depending on the reader's background, he or she may have difficulty in understanding how the phenomenon is described in a passage and how this information could be applied to an engineered system.

We found that caspase inhibitors did not affect cell death, although some caspase inhibitors that did not inhibit cell death impaired other stages in development and could block affinity-labelling of soluble extracts of Dictyostelium cells with an activated caspase-specific reagent.

Figure 4 Result from current biosearch using the engineering term, 'export' (Olie et al, 1998)

The needed translation is between two domain jargons, not full languages. Thus, the translation problem is focused on word substitution. The goal is to substitute biology-specific terms with language that is understandable to engineers. To do this, the algorithm must first identify a word requiring translation then find an appropriate term for substitution. In a practical sense, this translation needs to occur on text passages returned from a search term, and these translated passages must then be presented to the user.

To translate a jargon word, equivalent common words need to be found. WordNet is one possible source of substitute synonyms or hypernyms for the original target word. The goal is to substitute the target biological word with the most familiar word in the set of synonyms and hypernyms. The determination of the most familiar substitute term is based on BNC corpus frequency. Of note, some biological terms exhibit polysemy, so we have to determine the correct meaning of a target based on context in the original biological passage. To disambiguate the meaning of polysemic words, k-mean clustering based WSD is used, as discussed above.

Algorithm

This subchapter will detail the developed lexical substitution for biological terminologies. Pseudo code is in APPENDIX A.

Step 1: Generating potentially inspirational passages

Step 1 uses a version of a biological keyword search tool developed by Glier et al. (Glier et al, 2013). A term from the Functional Basis is entered into the program. The program then translates that engineering function into the equivalent biological functional terms via the engineering–biology thesauri (Chakrabarti et al, 2005; Cheong et al, 2011; Nagel et al, 2010). Next, the program performs a text search of the biological corpus. The biological corpora are texts from research journals and books about biology that have been compiled to support this and related research (Glier et al, 2013).

As an example, if the user were to input the engineering function *export*, the search engine would find every case of the word *export*, and the equivalent biological functional terms of *bind*, *block*, and so on. The list of synonyms for *export* is shown in Table 1.

Table 1 A part of University of Toronto Functional Set (Cheong et al, 2008)

Class (Primary)	Secondary	Tertiary	Correspondents
Channel	Export		Bind, block, breakdown, excrete, inactivate
	Transfer	Transport	Circulate, conduct, diffuse, pump
		Transmit	Communicate, transduce
	Guide	Translate	Synthesize, transcribe

Step 2: Identify words for translation

To determine what words to translate, the algorithm first identifies a "do not translate", or "ignore words", list. Two criteria are used to identify ignore words. First, the returned passages are compared to the 8000 most frequently used English words in the BNC. We have also included a list of family names occurring 100 or more times in the US (United States Census 2000) as part of the ignore list. Including this list of names is necessary in order to avoid attempts to translate the names of researchers often cited in the corpus. For example, a sentence clause such as "Brandy et al. pointed out" could be translated as "Alcohol et al. pointed out." The words not identified as in the ignore set are by default target words.

With the target words identified, there is some pre-processing on target words. The algorithm identifies compound nouns, single words, and abbreviations, as they are frequent and important elements of biological jargon. Multi-word compound nouns, such as 'Woronin body', need to be translated as such, not as two distinct terms. If a single word within the compound noun is translated first, the compound word is no longer

identifiable as biologically unique terminology. The first term, Woronin, is a common Russian family name. However, “Woronin body” means a microbody with a double membrane, which can be understood by an engineer. The algorithm recognizes a compound noun by the occurrence of two consecutive nouns. With target words categorized as compound, single, or abbreviation, the word level translation can begin.

Step 3: Processing the word in WordNet

In Step 3, the program gathers synonyms and hypernyms for the target word from WordNet. These words are candidate terms for lexical substitution. For a task such as translating biological terminology, sometimes a hypernym can deliver the concept of the original word better than any other synonyms. For example, in WordNet, synonyms of *Turbatrix* are *Anguillula*, *genus Anguillula*, and *genus Turbatrix*. However, a hypernym of Turbatix is *worm genus*. *Worm genus* is a good choice for comprehension by an engineer.

Identifying the correct hypernym is non-trivial, as immediate hypernyms are needed. Other hypernyms can be too broad. For example, an immediate hypernym of *Turbatrix* is *genus*. Substituting the word *Turbatrix* with *genus* in the translation would not provide rich enough information for the engineer to develop an understanding of the description contained in the passage. Developing an algorithm to select the best level of generality for a hypernym is beyond the scope of this research and remains for future work.

Subsequently, the algorithm processes monosemic and polysemic words. For monosemic words, the translation is based on substituting a target word with the most frequently used word among the set of synonyms and hypernyms. The familiarity of the word is measured by its frequency band in BNC. Each frequency band consists of 1000 words, with lower bands being more frequent. For example, band 3 contains the 2001st to 3000th most frequent words. For a single word, a synonym or a hypernym with the lowest frequency band will replace the original word. If there is not a suitable candidate for a monosemic target word, the definition in WordNet will be considered a translation for a target word. Compound nouns are processed in the same way that monosemic words are; however, the frequency band is determined by averaging the frequency bands of each word in the compound noun.

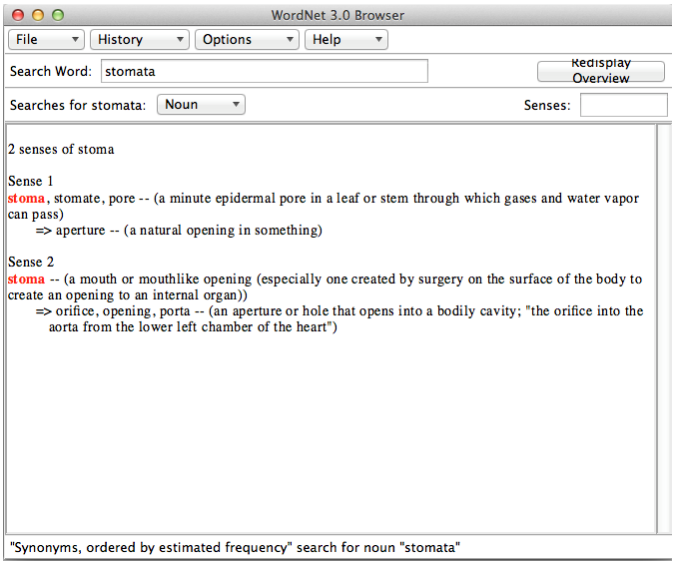


Figure 5 Multiple meanings of a biological term, 'stomata' in WordNet

For polysemic words, the WSD must be performed during the lexical substitution. For example, the word *stomata* in Figure 5 has multiple meanings and, thus, multiple translations. The correct meaning of the word 'stomata," and, thus, its correct substitution, may only be determined from the context in which it is used.

To determine word context, the algorithm first gathers keywords from the encapsulating sentence. The keywords occurring in the same sentence as the target word, which are used to disambiguate that target word, are referred to as the context window. Choueka and Lusignan found in their research that humans decide the meaning of a target word based on a few words surrounding it (Choueka & Lusignan, 1985). Banerjee and Pedersen extended this concept and applied it to computational WSD. Banerjee and Pedersen (2002) used four words – two words on the right side and two words on the left side of a target word – to solve the word disambiguation problem in their research (Banerjee & Pedersen, 2002).

Ignore words are excluded from the local context windows. Ignore words, also called stop words, are high frequency words such as *the*, *I*, *we*, and similar that are filtered before or after the NLP. These stop words are excluded from the problem of context determination as they do not help disambiguate target word meaning.

A global context window is a set of the most frequently used words from all the texts used in a substitution. In this algorithm, the global context keywords are the most frequent words in the passages generated from the functional keyword selected in Step 1. The reason for confining the global context window to text returned from the keyword search is that the biological corpus used in our search is very large, and it would be very

computationally expensive to run the algorithm, which counts the co-occurrence of target word and contexts, on the entire corpus.

Schutze developed an Expectation-Maximization WSD algorithm (EM algorithm) in his research and found that using a global context resulted in far greater accuracy than a local context (Schutze, 1998). Since both local and global context windows are important to disambiguate word sense, we use both contexts. Furthermore, Agirre and Rigau (1996) found that the accuracy of WSD converges after a context size of 25 words. For the present research, we use a global context of the 20 words that appear most frequently in the passage, as well as variable local contexts depending on the words surround the target word in the same sentence. As a result, we have a context window of over 25 keywords.

Once potential synonyms or hypernyms (Figure 6-a) and contexts for WSD are determined, the algorithm generates an NxM co-occurrence matrix (Figure 6-b). In the co-occurrence matrix, contexts for the target word are the column headings and synonyms or hypernyms for the target word are the row headings. The value of column–row intersection is the count of the context-synonym word pair occurring in the same sentence in the result passage. Each column can be considered as a vector that represents the feature of a synonym/hypernym of the target word. By grouping similar vectors, synonyms/hypernyms can be categorized. In other words, the program categorizes hypernyms and synonyms of a target word and binds similar elements into a cluster, using k-means clustering (Figure 6-c).

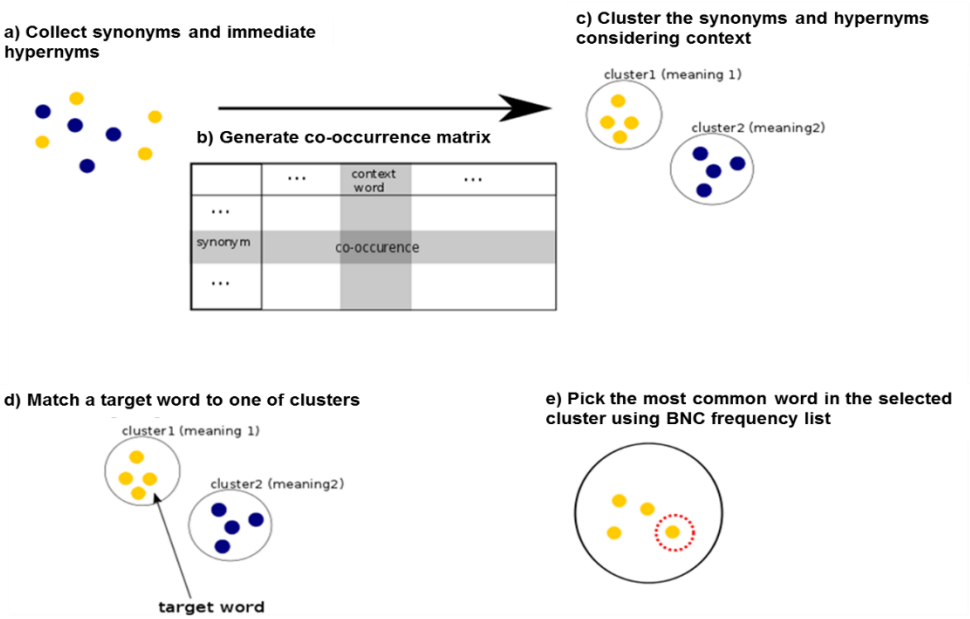


Figure 6 Clustering process of synonyms and hypernyms. Each dot represents a synonym/hypernym of a target word. Dots with the same color have the same meaning.

After the clustering process is finished, a target word, represented as a 1xM matrix, is assigned to one of the formed clusters, the program discriminates the meaning of the candidate words (Figure 6-d). Among the words in the selected cluster, the most common word according to the BNC list is selected as a substitution word (Figure 6-e). A summary of processing monosemic word and polysemic word is illustrated in Figure 7.

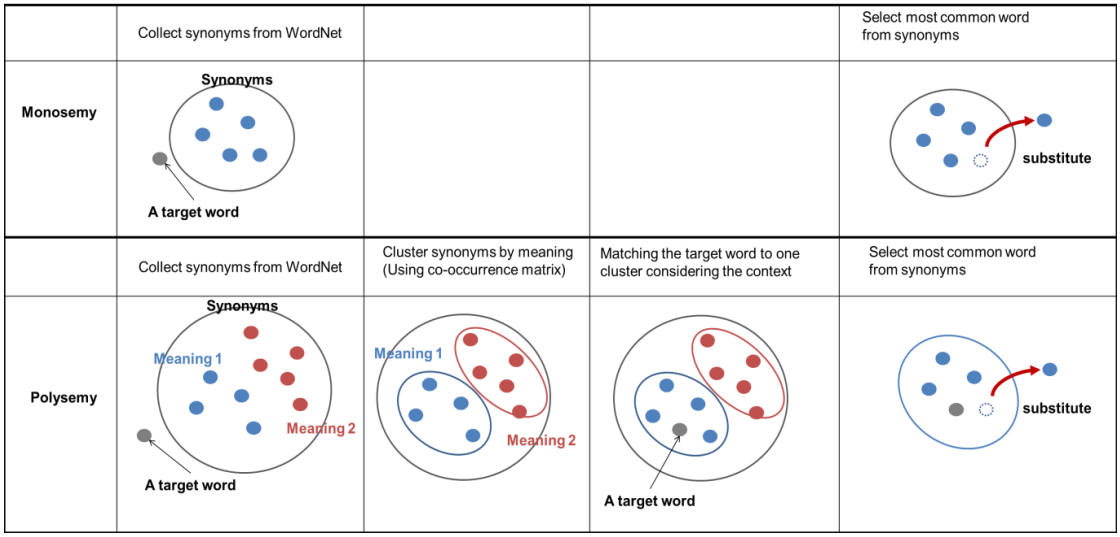


Figure 7 Comparison between processing monosemic word and polysemic word

Evaluation

The evaluation consists of two parts: evaluating the coverage of biological terms in WordNet and evaluating the effectiveness of the developed lexical substitution algorithm. Data selection, the baseline systems, and scoring criteria are described below.

Data selection

The test data used for the evaluation of the developed algorithm consist of 93 sentences with 127 unique target words. Each target words is not included in BNC 8000 list, and all sentences were manually selected as containing at least one biological term that has candidate words in WordNet. The passages were taken from the current biosearch tool's text database, mostly from biological journal papers (Glier et al, 2013).

To examine the WordNet's coverage of biological words, 4500 words were filtered by the developed algorithm. The target texts are also retrieved from a current keyword search (Glier et al, 2013). To categorize the filtered vocabularies according to their part of speech (POS), the parsing program, TreeTagger version 3.2, was used (Schmid, 1994). After the automatic tagging process by the parsing program, tagged POSs were again manually examined and corrected. Acronyms and abbreviated scientific words were manually collected and tagged in this procedure.

Baseline

For evaluating the system, two baseline systems were set:

1) Returns the most frequent candidate in the test documents
2) Returns the most frequent candidate based on BNC frequency

Especially, the 2nd baseline system has been set as a baseline for the most of lexical substitution systems (McCarthy & Navigli, 2007). The first baseline system was selected in this study considering the uniqueness of biological terms.

Scoring

Two annotators evaluated the validity of the developed algorithm. Both annotators were Ph.D. candidates in mechanical engineering. One annotator is a native English user and another annotator uses English as a second language. The scoring process of biological lexical substitution tasks is different from general lexical substitution task in several ways. First, unlike in general substitution tasks, annotators

often do not have knowledge about a target word. This means it is hard to find appropriate candidates for a target word in the absence of appropriate alternatives. Generally, the appropriate answer for a target word was suggested by annotators based on their own vocabulary; however, this approach is not viable in the present research. Second, sometimes a long descriptive statement can be a better substitution for a biological term than a simple one or two words. For example, to substitute the word "epithelium," it might be better to replace it with "membranous tissue covering internal organs and other internal surfaces of the body" than with "animal tissue," because a simple word, such as "membranous," sometimes cannot deliver important information. In this case, the annotators also cannot know suitable phrases that can replace a target word. Given these considerations, this research sets a golden standard by giving the annotators a candidate words list and letting them pick the suitable candidates among that list. The annotators were asked to score candidate words as 1 to 2. A score of 1 means a candidate is an appropriate substitute, and 2 means a word is somewhat suitable as a substitution in context but that there is a better substitute. If a candidate word is not suitable – for example has a different meaning from a target word in context – the score is left as blank. In addition, annotators are allowed to pick multiple answers for suitable words. All candidate words about which both annotators agreed were selected as a golden standard. If there was no agreement between the annotators, the words with the lowest average score were selected as golden standard.

After the golden standard is set, the best mode recall is calculated. This is one of the metrics generally used in lexical substitution tasks (McCarthy & Navigli, 2007) and can be expressed by the following equation:

$$\text{Mode R} = \frac{\sum_{sg_i \in TM} 1 \; if \; sg_i = m_i}{|TM|},$$

where sg_i is a system guess, TM is as a set of items in mode, and m_i is words in golden standard of i-th item.

Results

WordNet's coverage of biological terminologies

The validity of WordNet as a lexical source supplying candidate words was evaluated in the present study. Filtered biological terms from biological texts were categorized as in Table 2. The most noticeable feature of Table 2 is that WordNet does not cover the many biological terms filtered based on the BNC 8000 list. According to this research, 64.6% of biological terms are not contained in the WordNet database. More specifically, about 50% of nouns – except proper nouns (a location and a name of person) –, 70% of adverbs, 50% of adjectives, and 44% of verbs are omitted in WordNet. Further, most abbreviated scientific terms (ex. P. tigris) cannot be substituted using WordNet. Scientific acronyms, such as rsGFP (GFP red shifted), which also take a part of large portion of biological vocabularies, are also mostly excluded from WordNet. Discriminating these scientific acronyms is also very important in a biological terminology substitution problem considering their prevalence in biological terms (over 5% in this study). However, this problem is beyond the scope of the present study.

Another characteristic of biological terminologies is that there are many more monosemic words than polysemic words in biological terminologies. This characteristic can be logically understood, because most biological terms are domain-specific and rarely used in daily English, and thus are highly likely to have one or two uncommon meanings. This indicates that the lexical substitution for bioinspired design is more skewed to the simplification process of terms than the WSD process of polysemic words. Moreover, as we can see in Table 2, WordNet is not good for recognizing proper nouns (in this research, only location name or person's name are recognized). More than 76% of proper nouns are excluded from WordNet and could be recognized. Putting aside WordNet's coverage of proper nouns, BNC 8000 list is not good for filtering proper nouns. However, these proper nouns cannot be substitutes, and they should be recognized and excluded from a target words list to improve the substitution task. If the system tries to substitute proper nouns, the passages might have unnecessary explanations in text. Overall, the Table 2 shows the necessity of another lexical sources for supplying candidate words and filtering target words to improve the biological substitute task.

Table 2 WordNet's coverage of biological terminologies in current biosearch text database

	WordNet		Not in wordnet	Sum
	monosemy	Polysemy		
foreign language	1 (100.0%)	0 (0.0%)	0 (0.0%)	1
Adjective	233 (30.2%)	150 (19.5%)	388 (50.3%)	771
Noun	648 (30.2%)	310 (14.4%)	1189 (55.4%)	2147
Adverb	18 (20.7%)	6 (6.9%)	63 (72.4%)	87
Verb	36 (16.0%)	91 (40.4%)	98 (43.6%)	225
Preposition	0 (0.0%)	0 (0.0%)	1 (100.0%)	1
acronym	4 (0.6%)	9 (1.4%)	643 (98.0%)	656
proper noun	51 (14.1%)	35 (9.7%)	276 (76.2%)	362
abbreviated scientific term	0 (0.0%)	0 (0.0%)	241 (100.0%)	241
unit	2 (22.2%)	1 (11.1%)	6 (66.7%)	9
sum	993 (22.1%)	602 (13.4%)	2905 (64.6%)	4500

Evaluating the system

The evaluation result of the developed system is shown in the Table 3 and Figure 8. As many lexical substitution systems show (McCarthy & Navigli, 2007), lexical substitution task is not as easy as its general low precision and recall scores imply. Moreover, this result cannot be comparable to other lexical substitution tasks, mainly because the purpose, targeted vocabularies, and task data set are different.

However, its absolute numeric value is not very high, and the developed algorithm outperforms both baseline systems shown in Table 3. This result is mainly due to the developed system allowing the descriptive candidate (gloss in WordNet) while

baselines do not. As mentioned previously in the analysis of WordNet's coverage, most biological terms do not have various meanings and have a relatively easy WSD process compared to daily-use English words. The monosemic tendency of biological terms is closely related to their inclination to have little or no synonyms. In this case, it is hard to find a proper synonym because the number of candidate words is very small. Moreover, sometimes candidate words themselves are not proper for a substitution, because they are also domain-specific terms. In this case, if a terminology cannot be explained by one or two simple words, a phrase like a definition of a word is better. Actually, annotators tend to prefer descriptive substitutions over short candidates if candidate words are also biological terms. This is the main reason why the developed system scored better than the two baselines did.

Table 3 Percentage of the baseline systems and developed algorithm which returns answer in the golden standard

	baseline		System	Bio-terminologies
	docu_freq	BNC_freq		
Correct responses	36	42	60	128
Mode Recall (in %)	28.1%	32.8%	46.9%	100%

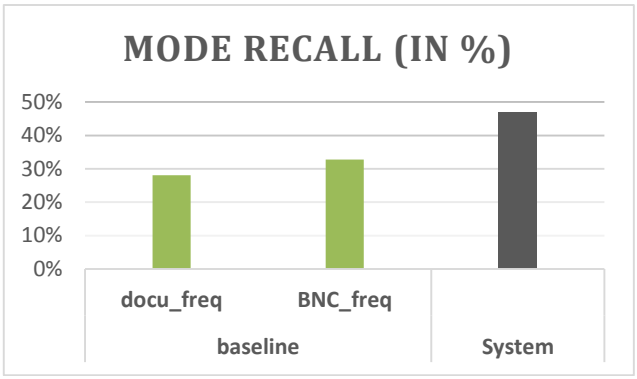


Figure 8 Mode recall of two baselines and the developed system

Conclusion

The WordNet based lexical substitution algorithm for bioinspired design has been developed and examined through this chapter. Biological terms have distinct differences from commonly used words, and these characteristics affect the lexical substitution task for bioinspired design. One conclusion is that because of the domain-specific tendency, many biological terms cannot be covered by WordNet, and thus require better lexical sources for supplying candidate words. In addition, the study shows that, in many cases, biological terms cannot be substituted by single words, and, thus, a translation task prefers a phrasal substitution to a single word to deliver the descriptive concept of a term for engineering designers.

Based on the rationale derived from this chapter, the next chapter will try to enhance the current lexical substitution algorithm using two other lexical sources: Integrated Taxonomy Information System (ITIS) and Wikipedia.

CHAPTER IV

LEXICAL SUBSTITUTION OF BIOLOGICAL TERMINOLOGIES USING WORDNET, INTEGRATED TAXONOMY INFORMATION SYSTEM, AND WEB-BASED LEXICAL RESOURCES

As delineated in the previous chapter, though WordNet is a well-known and widely used lexical source in computational linguistics (Fellbaum, 2010), it does not contain enough information for lexical substitution tasks with biological terminologies. Thus, the previous chapter showed the necessity of other supplementary lexical sources for improving the algorithm. This chapter tries to improve previous lexical substitution work by adapting the scientific name dataset, ITIS, and web-based lexical sources, Wikipedia and WordNik.

The main target of ITIS is scientific names of biological species. Especially, the common name list and the taxonomic hierarchy information included in its database will be used to recognize abbreviated scientific names. Other lexical sources, Wikipedia and WordNik, will process the words that cannot be processed with WordNet or ITIS. One benefit of those web based lexical sources is their variety and quantity of entries. As explained by (Ananiadou & McNaught, 2006), the features of biological terms show rapid growth of new terminologies and enormous variations in individual words. An open dictionary has less accuracy than that of a dictionary supervised by experts; however, its prompt response to newly generated terms makes it suitable to be used as a

lexical source for biological terminology processing. Noting this advantageous trait, open lexical sources, Wikipedia and WordNik, are added to the ITIS in this work.

To this end, first, we will go over the background work of the research. Then, the lexical substitution algorithm using combined lexical sources will be illustrated. The coverage of biological terminologies of each lexical source and the validity of new algorithm will be evaluated after the description of the algorithm. Finally, the last subsection will draw an overall conclusion of the work and suggest future studies.

Background

This chapter and the previous chapter share a lot of background knowledge, except information on the lexical sources used. Therefore, this subsection will be mainly dedicated to explaining characteristics of each lexical source.

Integrated Taxonomic Information System (ITIS) and Wikipedia

One of the biggest obstacles in biological text is the use of scientific names and taxonomic classification for biologists. Usually a scientific name is marked using binomial nomenclature, which consists of two parts: a species and a genus. For example, *Canis lupus* is the scientific name of the gray wolf, where *Canis* refers to a genus and *lupus* refers to a species. Genus names start with a capital letter, but species name should be all lower case. In addition, a scientific name is often represented in an abbreviated form (e.g., *C. lupus*).

This work tries to process these scientific names using the ITIS, which is a large biological name database containing taxonomic information of plants, animals, fungi, and microbes of the world, especially North America. The ITIS includes information such as full scientific name, hierarchy information, synonyms, vernacular names, and regional information of an organism (Integrated Taxonomic Information System). However, taxonomic hierarch information is provided with a Taxonomic Serial Number (TSN), and the full name is provided with TSN in a full scientific name list. By back and forth lookup of the hierarchy and full scientific name, the taxonomic hierarchy tree of the ranks include kingdom, phylum, class, order, family, genus, and species can be completed. Details about how our algorithm can find the easy alternatives of this scientific name using the ITIS will be described in the next subchapters.

To process a word not included in WordNet, the web lexical sources Wikipedia and WordNik are adapted in the work. Wikipedia is the well-known publically maintained web based encyclopedia (Wikipedia), known as one of the top five largest encyclopedia and dictionary websites. Though a publicly, or socially, maintained web based encyclopedia, Wikipedia has been used in computational linguistics and information retrieval studies (Adafre & De Rijke, 2006; Hassan et al, 2007; Milne & Witten, 2008).

WordNik is a web-based dictionary used in this work to support Wikipedia. One shortcoming of an encyclopedia such as Wikipedia for the lexical substitution task might be that the part of speech of entries is weighted towards the noun. Wikipedia provides a redirection link for a non-noun entry; however, not every non-noun can be covered by

this redirection scheme. WordNik contains over 6 million English words, and since it is a dictionary rather than an encyclopedia, its coverage of words other than nouns is expected to be better than that of Wikipedia. WordNet consists of multiple dictionaries including the Century Dictionary, WordNet, American Heritage Dictionaries, and Wiktionary. Wiktionary will be the main source of this work to back-up Wikipedia.

For implementation, the algorithm is written in Python using the Natural Language Toolkit (NLTK) (Bird et al, 2009). The detailed algorithm will be discussed in the next section.

Lexical substitution algorithm and implementation

The assumption made for the work is that the words not contained the WordNet can be considered as monosemic words. If we agree that WordNet only excludes rarely used words and that these rare words do not have many synonyms, this assumption gains justification. Thus, the word processed using ITIS, Wikipedia or WordNik will not require the WSD process, unlike the processing of words in WordNet.

This section details the process and methods of the substitution algorithm. The algorithm can be broken down into six major steps, as shown in Figure 9. In the basic sequence, these steps are: 1) to generate passages from a function keyword search; and 2) to identify words to not translate, then to preprocess and categorize words to translate according to suitable word substitution strategies. The words to translate, referred to as target words, are translated in steps 3), 4), 5), and 6), and steps 1) to 3) are as detailed in

the previous chapter. The text passage is then regenerated with translated terms and presented to the user. Pseudo code of the algorithm is attached in APPENDIX B.

Step 1: Generating potentially inspirational passages

According to the input keyword of an engineer, the algorithm returns the potentially inspirational passages. This step includes pre-processing of the filtered passages, including stemming and tagging.

Step 2: Identifying words for translation

In this step, the algorithm decides target words in the filtered text using BNC 8000. Since many prefix-attached compound words are understandable but not contained in BNC (e.g., *noncoding*) the algorithm tries to figure out if the compound word can be removed from a target word list. For this, the frequency band of a compound word and word removing a prefix are compared using BNC. For instance, in the above example, *noncoding*, the frequency band(s) of *coding* and *noncoding* are compared. If a word is not in the BNC list, its frequency band is set to 20. Since frequency band of *coding* is smaller than that of *noncoding*, the algorithm divides the word *noncoding* into *non+coding*, and decides if *coding* should be included in the target word. If the stem of a word, for example *coding*, still needs to be substituted, the result will be returned in the "*non+ (substitution of coding)*" format. The prefix included in this algorithm and frequency of each prefix among 1391 biological words is attached in APPENDIX C.

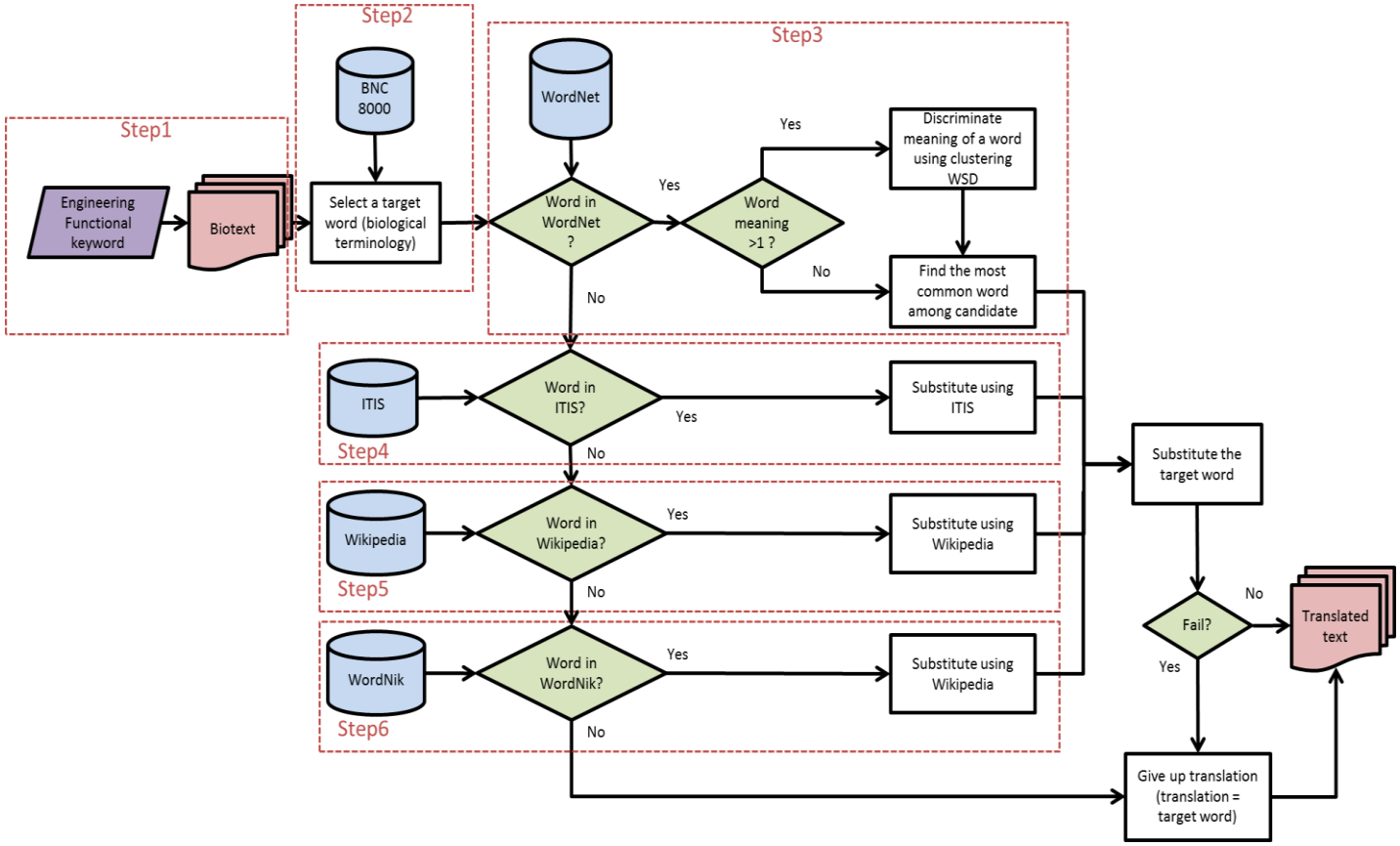


Figure 9 Passage translation and word substitution algorithm.

Step 3: Processing the word in WordNet

The processing of the word in WordNet as described in the previous chapter. Using clustering WSD, the algorithm discriminates the meaning of a word in context, and substitutes it with an easier similar meaning word.

Step 4: Processing scientific words

From this step, the difference between the previous algorithm and current algorithm arises. Step 4 mainly deals with a scientific name. A common, and important, aspect of scientific names is abbreviations, such as '*C. rubella*'. Abbreviations are typically not included in WordNet and Wikipedia. Thus, to enable the substitution of abbreviations commonly found in the biological literature, we include ITIS as a source.

First, it is required to find the possible full name of an abbreviated term. An abbreviated scientific name has a disambiguation problem, since an exact species name can be identified with the previous genus name. For example, '*C. sativus*' can be a species of a fungus (*Cochliobolus sativus*), a saffron (*Crocus sativus*), or a cucumber (*Cucumis sativus*). However, since the keyword search extracts the sentences containing the keyword from multiple biological papers, to find the full name, the algorithm goes back to the whole biocorpus and finds the nearest possible full name. Undisputedly, the possible full name can be searched by finding a word that starts with the first letter of a genus name and is followed by the matched species name.

The purpose of this step is to find the representative for an abbreviated scientific word. After finding a possible full name, the algorithm searches a common name for the

term using the ‘vernacular list’ in ITIS. If attempts to find a common name or genus fail, the algorithm finds a super-ordinated taxonomy using the ‘biological tree-hierarchy’ in ITIS and repeats the searching process until the algorithm finds a proper substitution. This step is illustrated in Figure 10.

For example, to translate “D. lacteipennis,” the tool finds the full name “Diclasiopa lacteipennis.” Since this name does not exist in the ITIS’s common name list, the program searches the biological hierarchy tree, and finds the upper-level biological classification terminology, “Diclasiopa.” The hierarchy tree of “D. lacteipennis” is in the lower portion of Figure 11. As “Diclasiopa” is also in neither the ITIS common name list nor Wikipedia, the program moves up a level in the classification scheme. This process continues until the program reaches the family name, “Ephydridae,” commonly called “brine flies,” and finishes the process.

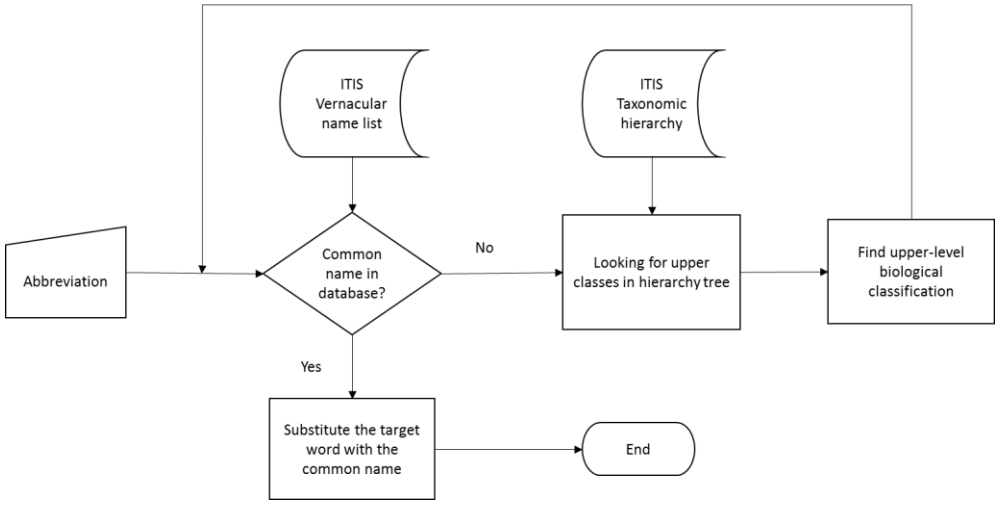


Figure 10 Specific procedures of Step 3

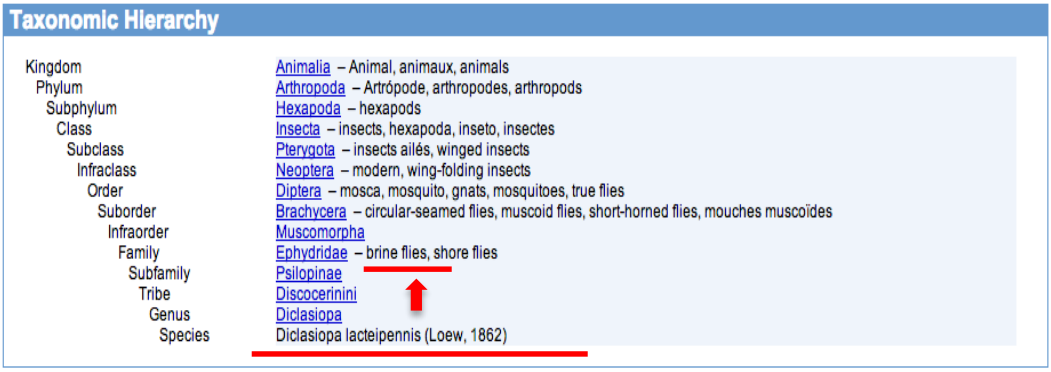


Figure 11 Taxonomic hierarchy of 'Diclasiopa lacteipennis' from ITIS.org (Integrated Taxonomic Information System)

Step 5: Using Wikipedia to find the definition of a target word

For words not found in WordNet and ITIS, the algorithms links to Wikipedia for substitution. Wikipedia does not contain explicit synonym or hypernym lists, but it does contain term definitions or explanations in a natural language format. To explain a word, natural language generally uses certain formats such as '*A* is a *B*' or '*A* is (also) called *B*'. Wikipedia defines words in a similar way. In Wikipedia, the definition sentence is typically located at the very beginning of the article, and uses the '*A* is a *B*' or '*A* is called *B*' format (or similar) to explain or define a term.

The substitution algorithm must recognize the definition sentence and extract the synonym or hyponym. To recognize the definition sentence, the algorithm uses key phrases that contain naming identifiers to identify and extract the definition sentence from Wikipedia. Table 4 shows the naming identifiers. These identifiers are determined

after a detailed investigation of 313 Wikipedia articles. In brief, a partial version of the substitution algorithm is run to identify jargon not found within WordNet. These terms are searched on Wikipedia, and the returned passages are read until the definition sentence is identified. Then, the naming identifier used in the definition is noted and logged. This process is repeated until the number of identifiers converges, as in the Figure 12. The five out of 313 articles do not have identifiers that define an entry directly. However, other than five articles, the number of identifiers converges to about 20. If one of these identifiers and a target word occur in same sentence, that sentence is considered as a definition sentence for the target word.

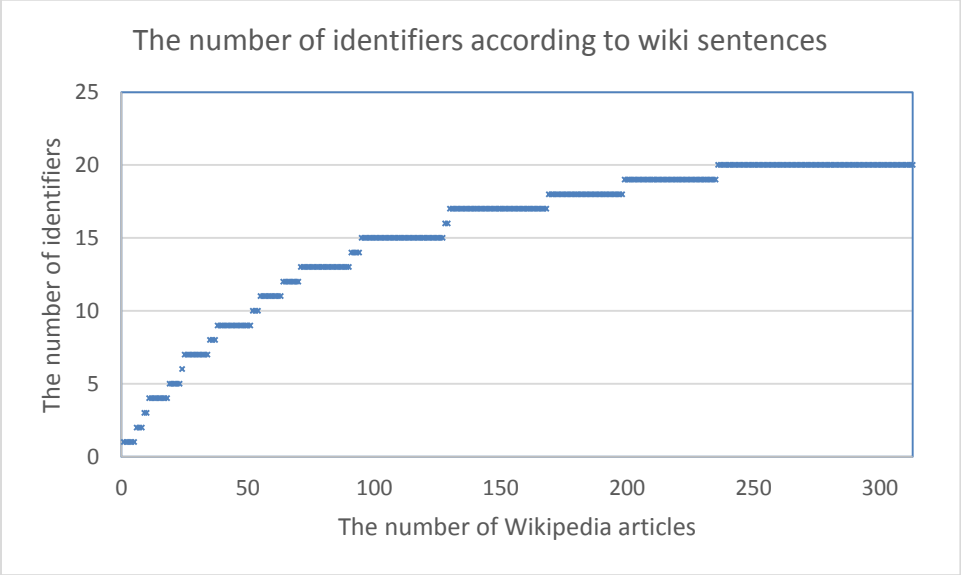


Figure 12 Plot of convergence of the number of identifiers

Table 4 The identifiers used to filter a definition sentence among an article from Wikipedia

Twenty identifiers heuristically found	is/are/was/were, (also) called, or, describe(s), (also) known as, refer(s) (to), constitute(s), represent(s), comprise(s), comprise(s), occur(s), called, mean(s), include(s), derived from, consist(s) of, be used in, involve(s), arise from, (be) defined by

Once the algorithm identifies a definition sentence, the algorithm finds the noun phrase (NP) in that sentence to replace, and clarify in meaning, the target word. To do this, the sentence is parsed and decomposed into a tree structure. Given a decomposed sentence, the noun-phrase near the identifier, and that on the other side of the target word in relation to the identifier, is considered the substitution for the target word.

An example is shown in Figure 13. The sentence is shown parsed and decomposed into a tree structure. In this sentence, the identifier is the verb, "is," and the target word is "Corallus." The NP on the opposite side of the target word is "a genus of non-venomous boas," as shown in the top of Figure 13. As a result, we can substitute "Corallus" with the phrase "a genus of non-venomous boas."

Where there are multiple identifiers in one sentence, multiple NPs can be retrieved, and the NP that has minimum average frequency band using BNC will be selected as a substitute. For words not included in the BNC, frequency band "20," a

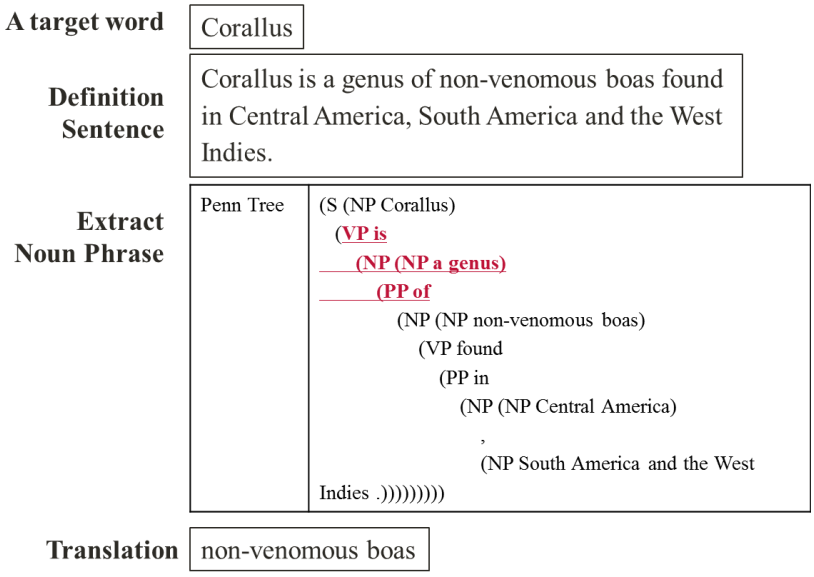


Figure 13 Tree-structure of a sentence, generated by parsing program (Link Grammar).

randomly chosen number, is applied to distinguish its frequency from that of words in the BNC 8000. Link Grammar, a syntax parsing program developed by Sleator and Temperley (Sleator & Temperley, 1991), is used to parse sentences. The algorithm uses the Penn Treebank sentence tree developed by Penn State University (Marcus et al, 1993).

Step 6: Processing the word with WordNik

As mentioned previously, articles titles of Wikipedia are usually nouns, and, thus, sometimes omit the non-noun entries. In addition, some articles do not contain a definition sentence that can derived from identifiers in Table 4. To process the words

that cannot be substituted for these reasons, they are processed with WordNik. Simply, the first definition in WordNik will be provided in brackets right after a target word. Words that remain un-substituted at this point (the end of Step 6) will be given up by the algorithm, and the original target word will be returned.

Evaluation set-ups

The evaluation consists of two parts: 1) coverage of the lexical sources, and 2) the validity of the lexical substitution algorithm. Simply, the coverage of Wikipedia and WordNik of biological jargon was evaluated using 1391 biological words that are excluded from WordNet according to the part of speech, and the coverage of ITIS for 241 abbreviated scientific terms was evaluated.

For the validity evaluation, two inter-raters were asked to evaluate 181 biological term substitution results from 102 sentences in two categories: the necessity of a substitution and the quality of the substitution. The results were generated using all four lexical resources. One inter-rater is a native English speaker and one inter-rater speaks English as a second language. Both inter-raters are mechanical engineering Ph.D. students. Usually for the lexical substitution task, inter-raters are asked to generate substitution candidates based on their knowledge, and with the golden standard made based on those candidates, precision and recall were measured (McCarthy, 2002).

Table 5 The rubric provided to the inter-rater to evaluate the system result

Rubric for validation of the algorithm <Necessity of a substitution> I cannot understand the original target word I kind of understand the target word but not clear about exact meaning of the word I can understand the target word and the substitution is not necessary <Quality of a substitution> The substitution seems have wrong meaning in context The substitution does not really bring the important information (ex. Too broad, "of pertaining to the *original word*") The substitution also has difficult words and makes the target word even confusing The substitution somewhat has difficult words, but give me idea about the context of the biological text The substitution helps me understand the biological text

However, for biological terms, it is hard to generate candidates because there are few or no candidates and, in many cases, the inter-raters lack background knowledge about biological jargon. Thus, rather than precision and recall, this work will measure the validity of the system based on a scaled standard. A rubric was provided to the inter-raters to evaluate the system result (Table 5).

Results and discussion

In this section, the evaluation of the ability of the algorithm and implementation code to translate passages is presented. The success and limitations of the algorithm are discussed to highlight areas for future research.

Coverage of biological terminologies

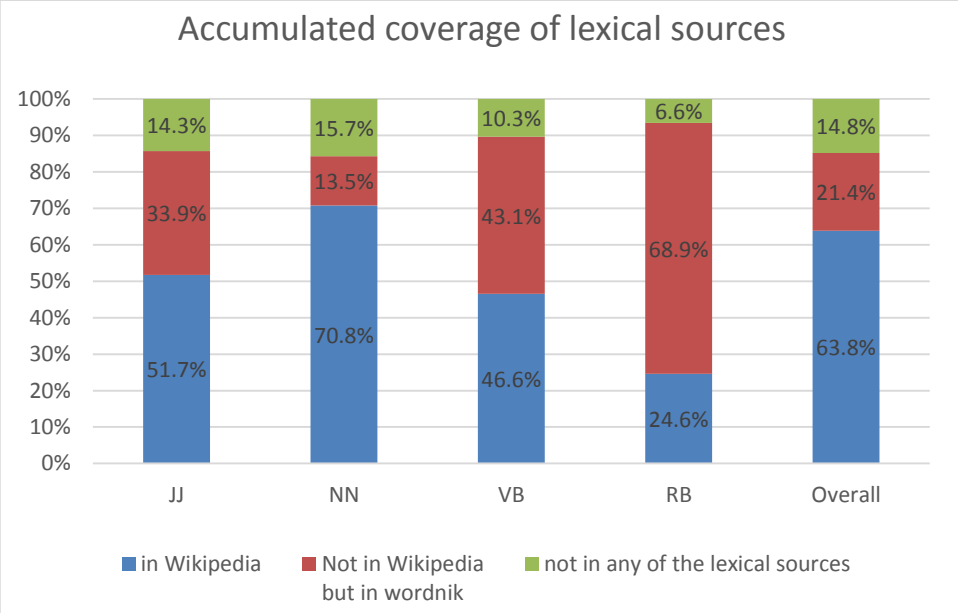
Having targeted 241 unique abbreviated scientific terms extracted in the previous chapter, we examine the coverage of ITIS, and we found that ITIS contains all 241 words in its database. This demonstrated that ITIS is a good lexical source for substituting the abbreviated scientific names contained in biological text.

On the other hand, the coverage of biological jargon (jargon not contained in WordNet) of Wikipedia and WordNik is shown in the Table 6. Since Wikipedia and WordNik consecutively process a target word, the accumulated coverage is more important than each individual source's coverage of words.

Table 6 Coverage of words not in WordNet according to lexical sources and POS.

	in Wikipedia	Not in Wikipedia but in Wordnik	Not in any of the lexical sources	sum
IN	1	0	0	1
JJ	148	97	41	286
NN	697	133	155	985
VB	27	25	6	58
RB	15	42	4	61
sum	888 (64%)	297 (21%)	206 (15%)	1391 (100%)

(IN:preposition, JJ:adjectives, NN:nouns, VB:verbs, RB:adverbs)



(IN:preposition, JJ:adjectives, NN:nouns, VB:verbs, RB:adverbs)

Figure 14 Accumulated percentage of terms in Wikipedia, Wikipedia+Wordnik, and terms neither in two lexical sources

As we can see in the Figure 14, Wikipedia is more focused on noun entries than other POS words, but with the help of WordNik, many more non-noun words can be processed. Especially, a lot of verbs and adverbs are omitted from Wikipedia but contained in the dictionary-type lexical source, WordNik. The combination of these two lexical sources can process 85.2% of biological terms excluded from WordNet. Since we found that WordNet has omitted almost 65% of biological terms, the entire coverage using three lexical sources, WordNet, Wikipedia, and WordNik, reaches almost 90% of biological terms, excluding acronyms.

Accuracy of the substitution

A key measure of usefulness of the algorithm is the accuracy of substitution: can the computer correctly identify the appropriate substitute word or words. Table 7 shows several results from the substitution using the input function “inhibit”. The table is organized into sentence pairs, and translated words are indicated with double square brackets.

The inter-rater agreement was measured using Krippendorff's alpha (Geertzen, 2012). The alpha values of the necessity substitution and the quality of substitution are 0.404 and 0.511, respectively, which indicate that agreement between the two inter-raters is moderate (Landis & Koch, 1977). The responses of the inter-raters are shown in Figure 15. As the figure indicates, target words selected using BNC 8000 often contain words that do not require the substitution task (44.2% and 28.2% from Figure 15). Especially, inter-rater1 showed even well acquainted words (44.2%) are more common than entirely strange words (40.9%), suggesting that an elaborated selection of target words is much-needed.

Table 7 Some results from biosearch, using the functional verb 'inhibit' (Queller & Strassmann, 2003; Suryavanshi et al, 2010)

Before Substitution	After Substitution
In our study, crabs could reduce interference such as kleptoparasitism by moving their victim to the other patch that has no competitor.	In our study, crabs could reduce interference such as [[The parasitic theft of captured prey, nest material, etc]] by moving their victim to the other patch that has no competitor.
If the strength of intra-specific interference is not so strong as to severely reduce pest control in the long-term, enemy species that engage in interference may be preferable to species that do not, especially in the case of species like Ooencyrtus that combines desirable properties such as high conversion efficiency and low mortality with undesirable ones such as a long handling time.	If the strength of intra-specific interference is not so strong as to severely reduce pest control in the long-term, enemy species that engage in interference may be preferable to species that do not, especially in the case of species like [[encyrtid wasps]] that combines desirable properties such as high conversion efficiency and low mortality with undesirable ones such as a long handling time.

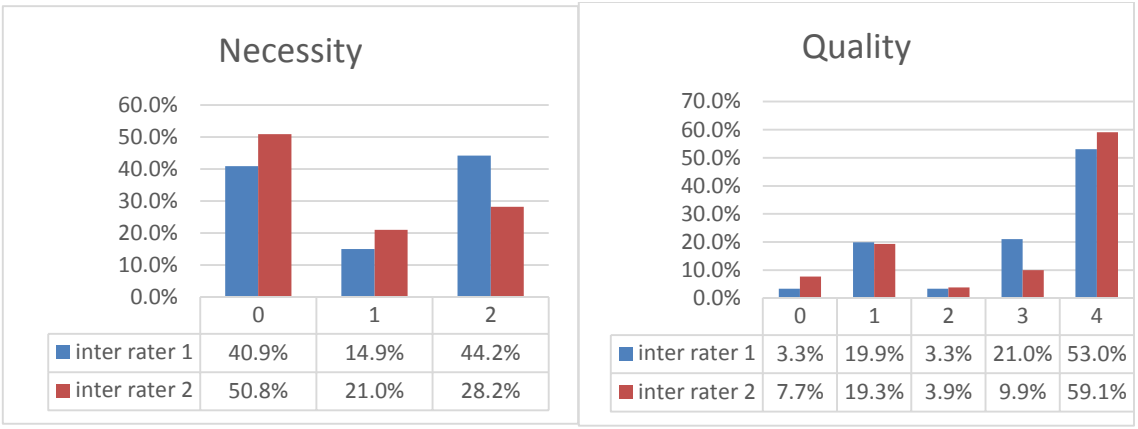


Figure 15 Responses of the inter-raters about 1) necessity of the substitution (left), and 2) quality of the substitution (right)

If we can consider quality metrics 3 and 4 as fairly good substitution results, we can simplify the 0-4 metric to "good substitution" and "bad substitution." Figure 16 shows the simplified measure of the substitution quality. Both good refers to a substitution result in which both inter-raters marked 3 or 4, and both bad indicates that both inter-raters answered 0–2 for the substitution. Not-agreed indicates where one inter-rater gave 0–2, but the other gave 3 or 4. Not-agreed results are due to the difference of vocabulary levels between inter-raters and considered a subordinated region to individual knowledge. Thus, we focus on both good and both bad answers. Encouragingly, among substitution results, 63% are answered as fairly acceptable substitution and expected to help engineers to understand the biological terms.

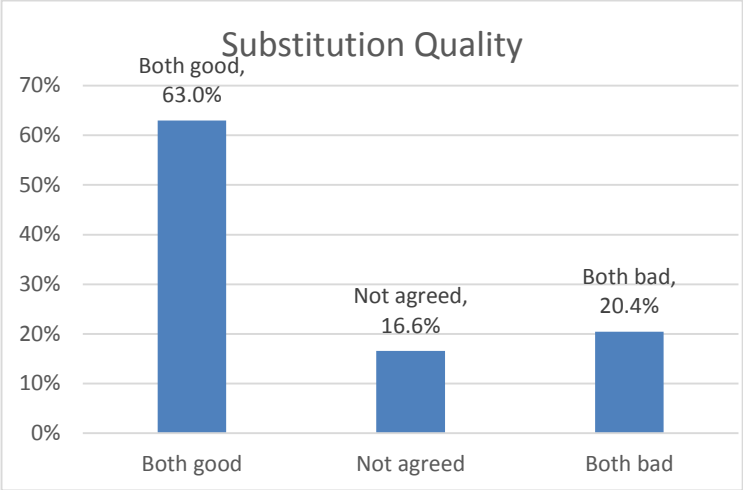


Figure 16 Simplified measure of substitution quality.

Analysis of limitations of the system

One reason for errors is due to polysemy words. Cases where the substitution failed to identify the correct meaning of an original word arose mostly from the failure of the K-means clustering WSD process. The quality value 0 indicates this type of error, which occupies 3.3 to 7.7% of answers in quality measure. Even though a large proportion of biological terms are monosemic words, these errors indicate that we need more accurate WSD process to improve substitution quality.

Though not as clearly an error, in some cases the immediate hypernym substitution of a target word results in too general a substitution. The general substitution leads to a vague meaning for the sentence and does not help in understanding the passage. In addition, results show that some biological terms are still not included in the corpora used in this research. This result indicates that current lexical sources are still

not perfect and we need more reference lexical corpora in addition to WordNet, Wikipedia, WordNik, and ITIS.

Other minor errors include the failure of retrieval of definition sentences or NPs. The former occurs when our identifiers are not applicable or when there is no definition sentence in an article. In addition, the extraction of NPs are largely dependent on Link Grammar, and the failure of tagging right tags can cause the failure of substitutions.

Case study: Improving solar thermal generation efficiency

To illustrate how the bioinspired keyword search and subsequent passage lexical substitution would support a conceptual design effort, a case study is presented here. The engineer's design activities using the developed tool is represented in the Figure 17. Briefly, a designer's activity consists of four steps:

- Analyze design needs/ system using functional basis
- Draw functional model using functional basis to decompose current system to its subsystems in order to better understand the problem
- Select one of the functional verbs in the previous functional model as an input keyword of the tool.
- Use the lexical substitution tool to find sentences, and try to get inspiration from biology

The design case used in this research is one that is widely studied in industry: improving the efficiency of a solar heater. For brevity, the case study here focuses on the

functional abstraction and the search for inspiration from biology to solve the problem of improving efficiency.

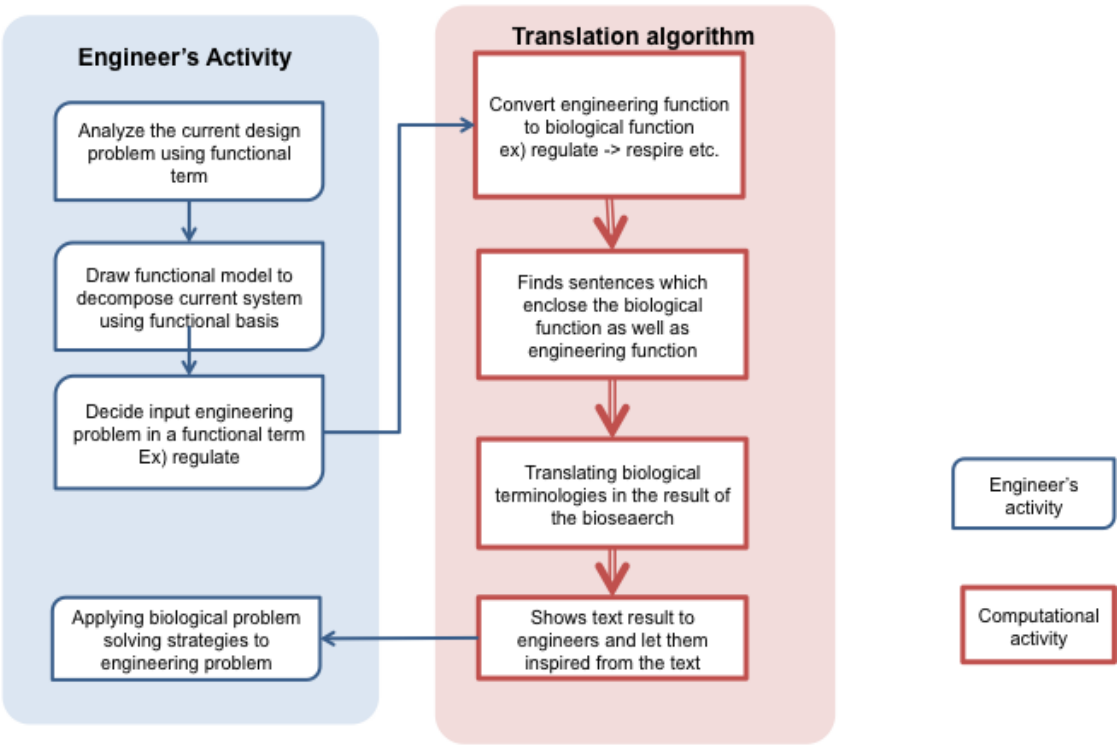


Figure 17 Engineer's design activity using the lexical substitution algorithm

Summary of the case study problem

Currently, solar energy is successfully utilized in many areas, such as solar cell technology, solar thermal energy generation, photovoltaic applications, and others. Compared to other solar energy technologies, solar thermal generation is important because of its relatively high efficiency and ability to store solar energy. The specific design problem focus of our case study is seeking methods by which we can improve the efficiency of solar thermal collectors.

Functional model for a current design

Figure 18 is a model of key functions of a solar panel. Focusing on improving the efficiency of a solar panel, we focus on the "collect" function. According to the Engineering-to-Biology Thesaurus, corresponding engineering keywords are *absorb, catch, breakdown, concentrate, digest,* and *reduce* (Nagel et al, 2010). The corpus is searched for these terms.

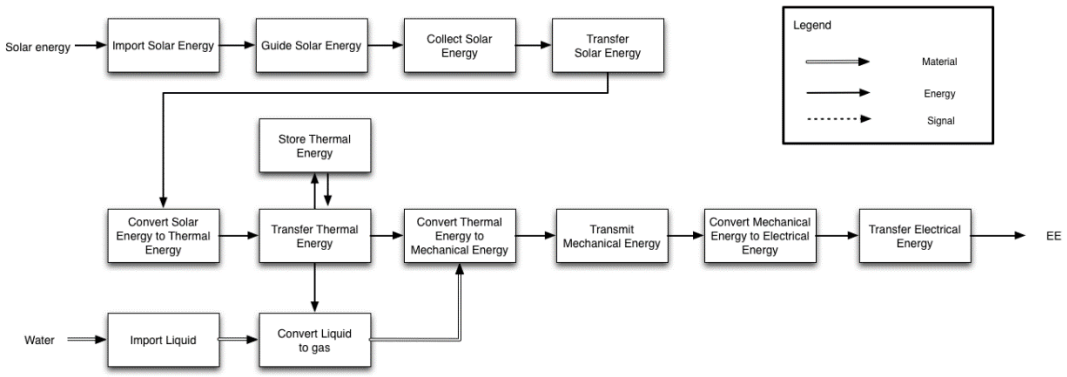


Figure 18 Functional model of a solar panel

Procedures of lexical substitution

The translation procedure introduced in Section 4 is detailed here for the given solar panel case study.

1) Generating potentially inspirational passages (Step 1 of Figure 9)

The first step of the algorithm is generating the inspirational text from the functional keyword selected by the designer, as illustrated in Step 1 of Figure 9. Table 8 shows part of the results using the function "collect." The Functional Basis term "collect" is converted to the biological functional terms, and the algorithm finds the passages containing these biological functional terms. In this discussion, we will focus on the passages returned for the biological keyword of *breakdown.*

Table 8 Selected result from the bio-search tool, using functional keyword 'collect'(Jackson et al, 2005; Mogilner & Keren, 2009)

Function Verb : Absorb

Before Translation:
Recent accounts of optic ataxia based upon electrophysiological recordings in monkeys have proposed that this disorder arises because of a breakdown in the tuning fields of parietal neurons responsible for integrating spatially congruent retinal, eye, and hand position signals to produce coordinated eye and hand movements (Banerjee & Pedersen, 2002).

After Translation:
Recent accounts of optic [[nervous_disorder]
] based upon [[electrophysiological]] recordings in monkeys have proposed that this disorder arises because of a breakdown in the tuning fields of [[parietal]] [[nerve_cell]] responsible for integrating spatially [[congruent]] [[pigment]], eye, and hand position signals to produce coordinated eye and hand movements [1].

##############################
Before Translation:
Matrix breakdown is the limiting step in this type of migration [91], so perhaps cells switch to amoeboid motility when the nucleus can squeeze through the malleable pores.
After Translation:
Matrix breakdown is the limiting step in this type of migration [91], so perhaps cells switch to [[single-celled life-forms characterized by an irregular shape]] [[movement]] when the nucleus can squeeze through the [[tensile]] pores.

2) Identifying words for substitution (Step 2 of Figure 9)

In Table 8, highlighted words "ataxia," "electrophysiological," "parietal," "neurons," "congruent," and "retinal" are not in the BNC 8000 and, thus, are identified by the algorithm as words difficult to understand by an engineering audience. These terms become the target words for substitution.

3) Processing abbreviations (Step 3 of Figure 9)

If there is an abbreviation in the result, the algorithm translates the term using the ITIS corpus. Here, "C. rubella" is translated to "bessbug," and "L. perenne" is translated to "blue flax" using ITIS corpus.

4) Finding the synset using WordNet (Step 4 of Figure 9)

Next, the algorithm translates the term "ataxia" from Table 8. Since ataxia is in the WordNet database, the algorithm tries to find its synonyms and immediate hypernyms. Synonyms of ataxia are "ataxy," "dyssynergia," and "motor ataxia," and immediate hypernyms are "nervous disorder," "neurological disorder," and "neurological disease."

5) Processing monosemic and polysemic words (Step 5 of Figure 9)

Ataxia is an example of a monosemic word. The algorithm finds the average frequency bands for each of its synonyms and hypernyms using the BNC frequency list, as explained in Section 4.5. After comparing the frequency bands, the hypernym "nervous disorder" is chosen as a substitution for ataxia (because it has lowest average frequency band among all possible candidates for substitution).

To translate the polysemic words in Table 8, such as the word "malleable," the algorithm first gathers every synonym and immediate hypernym without discriminating their meanings. Malleable has two meanings, as shown in Figure 19.

After gathering all target word substitutes, the algorithm groups the potential substitutions according to their meanings and contexts. Words such as “ductile” and “malleable” might be in the same cluster, while those such as “pliable,” “pliant,” “tensile,” and “tractile” might be in another cluster, if we assume that we neglect the overlapped synonyms. Using the WSD technique introduced in chapter 4.5, the word malleable, as it appears in Table 8, matches the later cluster according to its context. Then, the algorithm selects the most frequently used word from the appropriate cluster as a substitution for “malleable” using the BNC frequency list, which, for this example, is “tensile.”

6) Using Wikipedia for lexical substitution (Step 6 of Figure 9)

The term “amoeboid” is not contained in WordNet. Thus, Wikipedia is mined for a suitable substitution. Using Wikipedia, “amoeboid’ is translated into “[[single-celled life-forms characterized by an irregular shape]]” as shown in Table 8.

Malleable:

1) (adj) ductile, malleable (easily influenced)

2) (adj) ductile, malleable, pliable, pliant, tensile, tractile (capable of being shaped or bent or drawn out) "ductile copper"; "malleable metals such as gold"; "they soaked the leather to made it pliable"; "pliant molten glass"; "made of highly tensile steel alloy"

Figure 19 Two meanings of the word 'malleable'

Results from the Bio-search tool

After reviewing the search results, several methods are suggested by nature. A designer might try to dye a surface of a solar panel or mimic the structure of eyes to improve efficiency. Of note, independent both of this case study and of the specific passage returned here, some scientists have suggested the possibility that the non-reflective nanostructure of a moth's eye can be applied to the solar device to enhance efficiency (Parker & Townley, 2007). According to this study, the nipple layers of a moth's eye can prevent light from reflecting off its surface, thus offering potential structures for consideration in improving the efficiency of solar cell technology. In another study, scientists tried to find new dyeing materials in nature in attempts to improve solar cell technology (Calogero & Di Marco, 2008).

Conclusion and future work

For creative design, the developed system serves as an effort to provide biological textual inspiration to engineers. Bioinspired design research groups have tried to develop methodologies that can stir up the analogical inspiration of a designer through bio-textual representations. Those efforts usually have been accomplished by establishing the thesauri, the word set connecting engineering and biology terminologies based on functional bases. Engineering-to-biology thesauri can help to reduce the gap between biological terms and those of engineering; however, they still cannot make biological texts completely understandable to engineers. Therefore, in an effort to deliver effective analogical transfers based on textual inspiration, an algorithm translating the biological terminologies is introduced in the present research. The fundamental principle of this algorithm rests in using WordNet to find simple substitutions for complex biological terms. For polysemic words, the program solved an ambiguity that exists in discerning a word's meaning by using clustering WSD. Additionally, the algorithm uses the ITIS corpora, Wikipedia, and WordNik to translate the biological terminologies that are not contained in the WordNet database. Case studies suggested the practical usage of this tool and examined its validity as an inspirational source.

The lexical substitution is important because it can help to bridge the gap between two different fields of research – in this study, those of biology and engineering. The developed algorithm can reduce the burden of engineers as they study unfamiliar areas in hope of finding biological inspiration in textual resources. In other words, the transfer of analogical solutions from biological systems to engineering

problems becomes an easier process than it would otherwise, without the lexical substitution, be. This process is made possible by the application of lexical substitution theories to bioinspired design. Bioinspired design studies have already tried to adapt many NLP theories from computer science to the fields of engineering in order to find useful lexical bridges between the studies of biology and engineering. The present study is meaningful because it applies lexical substitution theories to biological design studies for the first time. The authors hope that this research can foster additional momentum toward interdisciplinary studies of bioinspired design and other fields in the realm of computer science.

Further work remains for the future. Since WSD, used as a main algorithm in the study, is still an open question in the field of computer science, improvements to the program are anticipated as WSD technologies evolve. The program, or algorithm, uses K-mean clustering because of its efficiency; however, more accurate and faster WSD methods may be applied to the design process in future. The inefficiency that resulted from the task of combining multiple NLP programs into one algorithm is also required to be resolved.

Further utilization and expansion of this algorithm also remains for further study. One of the prominent future works, which can be derived from this research, is building parallel (Engineering–Biological) texts. The translated result has limitations, because the developed algorithm is not based on any such parallel texts. Since parallel texts enable the training of the computer, which generally results in a better LS rate, using solely monolingual sources for the lexical substitution leads to less accurate substitutions than

does using parallel texts. However, based on this study, researchers can start building parallel texts between the two different domains, and this will result in better accuracy for future studies.

This research dealt only with the lexical substitution algorithm itself. Observation of the results generated after implementation of this lexical substitution in a design process remains incomplete. As a consequence, it is hard to tell that what kind of effect is inherent within the design cognitive process due to this lexical substitution methodology, and this needs to be examined. Future human experimentation on biological inspirational processes using translated texts can be conducted to help researchers explore the impacts and implications of the algorithm. However, those questions are beyond the scope of the current work.

CHAPTER V

CATEGORIZING BIOLOGICAL INFORMATION BASED ON FUNCTION-MORPHOLOGY FOR BIOINPSIRED CONCEPTUAL DESIGN

As introduced earlier, on top of the lexical gap problem, another key problem of keyword search is the large volume of text passages returned. In addition, many of the returned passages do not contain specific biological solutions in the text. This means engineering designers have to invest a huge amount of time to read all the passages to find a biological solution, or, moreover, have to read information similar to that already determined not useful in multiple passages. Though this may sound like a minimal inconvenience; however, in practice it prevents effective use of the keyword search method. The goal of the designer is to find inspiration in a biological system, not sift through a large volume of unrelated biological information.

The core research objective of this chapter is to identify solution morphologies as found in the keyword-identified passages and cluster the biological solutions based on them. Thus, only passages containing morphological terminology are returned to the user. Additionally, passages are returned to the designer focused on solutions buried in the text, which augments inspirational processes of the designer. To extract information related to structure or shape, referred to as morphology in this research, in biological text, this research built on the WordNet noun and morphology categorization. Based on the morphological solution, LSA (Landauer et al, 1998), which is widely used to text

clustering in information retrieval area, will be adapted to classify biological returned passages into clusters describing a similar morphology.

Background

The work presented here builds on some representational schemes from the design community and some related computational approaches. Here, this background work is presented in detail.

Morphology

Morphology can indicate two kinds of meaning. In linguistics, morphology is a study regarding the form of morpheme or other linguistic units. However, this research only uses the morphology in a biology or engineering sense. The Oxford English Dictionary defines the word "morphology" as "The study of the forms of things" and "A particular form, shape, or structure." Our focus on morphological elements of a physical process, entity, or solution in this research is due to the importance of morphology in function-based conceptual design as well as its frequent usage in biological texts. The benefit of identifying the key biological morphology that enables the execution of some biological function is that it better allows the engineer to understand the biological system and how it works from an engineering perspective. Gero and Kazakov (1999) and his colleagues, pointed out that enlarging the structural design space can be also considered a form of creative design. Although structure does not explicitly imply a specific function, and vice versa, if a designer learns a relation of function and structure, it is very hard to discard that studied knowledge (Qian & Gero, 1992). As a result, if a

designer can introduce a totally new structure and form into a conceptual design effort, it can enlarge the conceptual design solution space, thereby contributing to creative design. Likewise, importing biological structure, or morphology, into engineering design is expected to help engineers discover and create unique morphologies not found in conventional engineering designs. Thus, the research presented here aims to mine morphological solutions in the text passages returned by biological keyword searches. The detailed research approach to discover morphological expressions in the text will be described in a later section.

Latent Semantic Analysis (LSA)

This research uses LSA to cluster the results of the biological keyword search. As explained previously, the keyword search produces a large number of brief text passages containing the functional keyword. The aim is to cluster and organize these results in some coherent fashion before presenting them to the searcher. LSA is a widely used automatic information retrieval technique, created by Landauer et al. (Landauer et al, 1998). It uses singular value decomposition (SVD) to extract relations between documents based on words. According to the explanation of Laundauer (Landauer et al, 1998), LSA can be mainly viewed two ways, the "expedient" to estimate the relatedness of a word to a larger text unit, or a computational model that is fundamental to obtain or use knowledge (Landauer et al, 1998). In many cases, we might not separate the two point of views as does this research, which tries to find similar morphological noun groups as well as to mine morphological solutions in documents. In detail, LSA is also

often used in automatic text (or document) clustering or topic identification, as it has been shown that there is a close correlation between SVD categorization and human judgment-based categorization (Laham & Foltz, 1998).

The very important analogy of LSA and human rationale for processing knowledge is utilizing latent (or already known) information to understand and process new information. In LSA, this can be achieved by the mathematical theory, SVD. The sparse data has a high dimensional space. SVD rotates and translates the dimension axes and tries to find dimensions that explain characteristics of data well. During SVD, we can get eigenvectors that represent the importance of each dimension in the descending order. While reducing less important dimensions, data is projected to the lower dimensions. This mapping process reduces the noise and enables us to find latent relations between words and documents.

SVD decomposes an original matrix into three matrices. These three matrices are two orthogonal matrices and one diagonal eigenvalue matrix. For the language model, a term-document (or any text segment such as passage or sentence) matrix is typically used for LSA. In the matrix, each row stands for a unique word, each column stands for a document, and the intersecting cell entry is the frequency of the word (row) as it appeared in the passage or document (column). The term–document matrix will be decomposed into three matrices: term vector matrix, diagonal eigenvalue matrix, and document vector matrix using SVD.

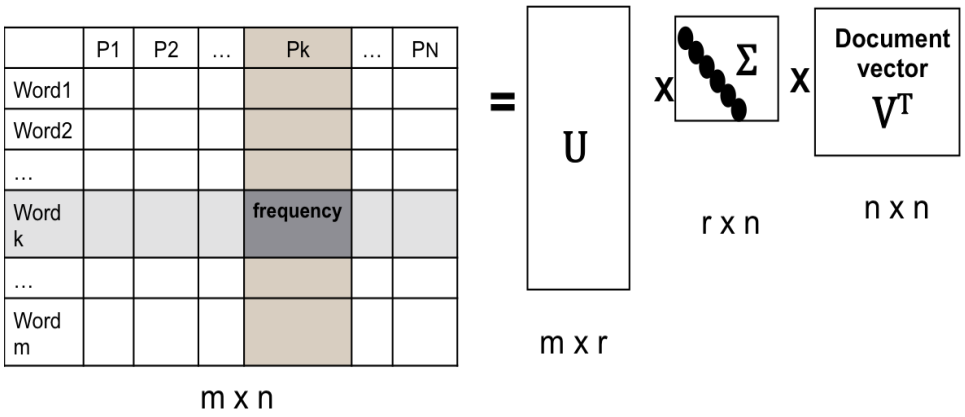


Figure 20 Term-document matrix and its decomposition to three matrices using LSA

If we refer to the m (word) x n(passages) dimensional matrix as C, SVD expresses the matrix C as follows:

$$C = U\Sigma V^T,$$

where U is an m x m matrix, Σ is an m x n diagonal matrix whose diagonal elements are eigenvalues, and V^T is an n x n matrix (Manning et al, 2008). The SVD matrix decomposition is illustrated in Figure 20.

Landauer et al.(1998) pointed out that the LSA finds the correlation of words and documents by contrasting differences between data, especially using the information that certain words does not occur in certain documents.

As mentioned before, reducing the dimensions in data is essential to the SVD process. For large data sets in which m is much larger than n, a truncated SVD can be used with improved computational efficiency and limited loss of data resolution. A

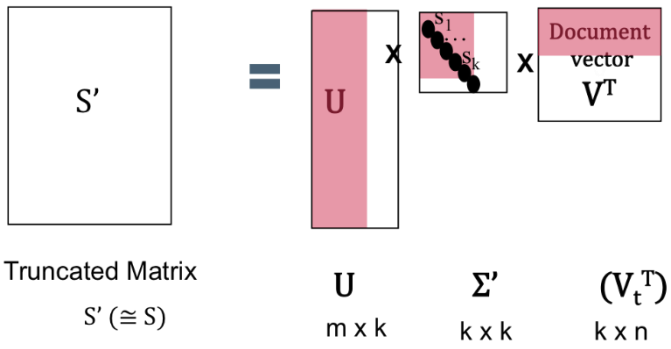


Figure 21 A Truncated SVD

truncated SVD zeros out the eigenvalues in Σ, except the k largest singular values, and removes the columns and rows of U and V_t according to the dimension of new diagonal matrix Σ'. The truncated SVD generates the truncated term matrix, U_t, and truncated document matrix, V_t, respectively. We can recalculate the document matrix to k dimensions by multiplying Σ' and V_t^T, and the result of this multiplication is document vectors mapped to k dimensions. Likewise, multiplication of the truncated term matrix U_t and new diagonal matrix Σ' represents k-th dimensional mapped term eigenvectors. The truncated SVD matrix C', generated by multiplying U_t, Σ', and V_t^T, approximates the original matrix C showing semantic relationships between elements by approximate input data (Landauer et al, 1998). A truncated SVD is illustrated in Figure 21

Deciding dimensionality of SVD is also an important factor in LSA because reducing dimensions can remove noise but also lose some information. The appropriate number of singular values is still heavily studied in different areas and remains an area of debate and discovery (Bingham & Mannila, 2001; Globerson & Tishby, 2003;

Landauer et al, 1998). It is widely accepted that reduced dimensions between 200 and 600 are acceptable for large corpora, but not for small corpora such as an essay (Villalon & Calvo, 2009) or text corpora such as that used in this study.

Weight functions of LSA

LSA can directly use term-document vectors and apply SVD to those vectors to find the relation between terms and documents. However, for the purpose of improving accuracy, in many cases, a weight function is applied to term-document vectors before SVD is applied. A co-occurrence matrix is transformed by weight functions before applying SVD in the LSA process (Nakov et al, 2001).

A weight function takes into consideration the importance of a word to a document and to a collection of documents. Thus, weight functions usually consist of two parts: a local and global weight function. Assume that the weight function of term i in document j is $w(i,j)$, then the weight function can be expressed as the following (Nakov et al, 2001):

$$w(i,j) = L(i,j)G(i)$$

where $L(i,j)$ refers to a local weight function and $G(i)$ refers to a global weight function.

Local weight functions are usually proportional to term frequency in a document, and global weight function reflects the importance of a term in a set of documents (Nakov et al, 2001).

Various weighting schemes have been tested in previous studies (Nakov et al, 2001; Salton & Buckley, 1988). Among many weighting schemes, TF-IDF is a common weighting function. Term frequency (TF) literally indicates the number of term occurrences in a document. It is the same as the vector values used in the co-occurrence matrix. Inverse document frequency (IDF) is a global weight function, which is defined as,

$$\text{IDF}(t_i) = log \frac{N}{df(i)},$$

where t_i, $df(i)$, and N represent the i-th term, the number of documents containing term t_i, and the total number of documents, respectively. Thus, IDF suggests that as a term occurrence increases across all documents increase, its importance in the document collection decreases. In other words, if a certain word appears often in every document, the importance of that word can be considered as non-significant. Such a weighting function will prevent common words, such as articles, from being considered the most important words in a document collection.

The Logarithmic term frequency is a transformation of TF, and it is represented as $\log(\text{TF}(i,j) + 1)$. This study will use a combination of logarithmic TF and IDF, since this approach has shown reasonably good accuracy in previous studies (Nakov et al, 2001; Wild et al, 2005).

In summary, the weighting function used in this study is represented as follows:

$$w(i,j) = \log(\text{TF}(i,j) + 1) \times log \frac{N}{df(i)}.$$

Expectation-Maximization (EM) algorithm

Even though LSA is considered as light clustering, the Expectation-Maximization (EM) algorithm will be applied to the document. Before starting the EM algorithm, K-means clustering can be used as an initialization step in the EM algorithm. It generates a random seed centroid of clusters and repeats the iterative process that consists of data assignment and recalculating centroids of clusters. In this iterative process, each data point is assigned to the nearest centroid, which minimizes the Residual Sum of Squares (RSS), defined as $\sum_{i=1}^{k} \sum_{\vec{x} \in S_k} \|\vec{x} - \mu_i\|^2$. Here, $\vec{x}$, μ_i and S_k refer to a data vector, a mean of i-th cluster, and the set of k-th cluster, respectively. Then, a new centroid is recalculated as a means of data points in a set. The process repeats until the stop criterion is met (Manning et al, 2008).

Our main clustering algorithm, the EM algorithm is a statistical model used to estimate hidden parameters in a set of data with iteration, and it got its name from the study of Dempster et al. in 1977 (Dempster et al, 1977). Briefly, the EM algorithm tries to find the hidden or missing parameter by finding maximum likelihood from a given data set. The EM algorithm consists of two steps, the E- step (expectation step) and M-step (maximization step). The E-step estimates the expected hidden variables from the current model; and the M-step finds new model which maximizes the log likelihood about a given model and currently estimated hidden variables. The research uses Gaussian mixture model for EM algorithm. Gaussian mixture model assumes that data consists of k gaussian and tries to find the mean and covariance which explains the data most.

The explanation for the EM algorithm here is based on the paper by Bilmes (Bilmes, 1998). If an observed data set is $\mathbf{Y} = \{Y_1, Y_2, Y_3, \dots, Y_n\}$, the log-likelihood function for $\mathbf{Y}$ can be expressed as $\log p\,(\mathbf{Y}; \theta_0)$ where θ_0 is unknown parameter. If we put missing data as $\mathbf{U} = \{U_i\}$**,** our log-likelihood function the set of (Y_i, U_i) becomes

$$\mathcal{L}_n(\mathbf{Y}, \mathbf{U}; \theta) = \log f(\mathbf{Y}, \mathbf{U}; \theta) = \log f(\mathbf{Y}; \theta) + \log f(\mathbf{U}; \mathbf{Y}, \theta)$$

by the chain rule. The expectation of $f(\mathbf{Y}, \mathbf{U}; \theta)$, which is defined as $Q(\theta_*, \theta)$, is known for the best estimator for the log-likelihood of $\mathcal{L}_n(\mathbf{Y}, \mathbf{U}; \theta)$. We can write expectation of log-likelihood as below,

$$Q(\theta_*, \theta) = E(\log f(\mathbf{Y}, \mathbf{U}; \theta | \mathbf{Y}, \theta_*))$$

EM- algorithm obtains unknown variable θ for each step which maximizes the Q by given value Y, and estimated hidden value θ_* for each step in EM algorithm.

In Gaussian model, the objective of EM algorithm is estimating the hidden variable, in this case, mean μ_j, covariance σ_j, and the probability that each data belongs to j-th gaussian. If we know that there are k Gaussian (clusters), $S_1, S_2, \dots, S_k$, we can define the Gaussian distribution as

$$f_j(Y_i; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(Y_i - \mu)^2}{2\sigma^2}}$$

And our log-likelihood can be expressed as

$$\mathcal{L}_n(\mathbf{Y};\theta) = \prod_{i=1}^{n} \log p(\mathbf{Y};\theta) = \sum_{i=1}^{n} \log \sum_{j=1}^{k} p_j\, f_j(\mathrm{Y_i};\mu,\sigma)$$

$$= \sum_{i=1}^{n} \log \sum_{j=1}^{k} \left(p_j \frac{1}{\sigma_j\sqrt{2\pi}} e^{-\frac{(Y_i-\mu_j)^2}{2\sigma_j^2}} \right)$$

where μ_j and σ_j refers to mean and deviation, and variable p_j refers the membership probability of j Gaussian (cluster).

However, as we can see in the above log-likelihood function, it is extremely difficult to maximize the log-likelihood function. So, random variables δ_i is introduced, where $\delta_i \in \{1,2,\dots,k\}$.

$$\mathrm{P}(\delta_i = j) = p_j$$

$$\mathrm{f}(\mathrm{Y}_i = \mathrm{y}|\delta_i = j) = p_j$$

$$p_j \geq 0, \quad \sum_{j=1}^{k} p_j = 1$$

The log-likelihood function of $\{(\mathrm{Y}_i, \delta_i)\}$ is

$$\mathcal{L}_n(\mathbf{Y},\boldsymbol{\delta};\theta) = \sum_{i=1}^{n} \log \sum_{\delta_i=1}^{\delta_i=k} p_{\delta_i} f_{\delta_i}(Y_i;\theta_{\delta_i})$$

Then, our expectation of log-likelihood become

$$Q(\theta_*,\theta) = E(\mathcal{L}_n(\mathbf{Y},\boldsymbol{\delta};\theta)|\mathbf{Y},\theta_*)) = E(\log f(\mathbf{Y};\theta)|\mathbf{Y},\theta_*) + E(\log f(\boldsymbol{\delta};\mathbf{Y},\theta)|\mathbf{Y},\theta_*)$$

$$= \sum_{i=1}^{n} E(\log p_{\delta_i}|Y_i,\theta_*) + \sum_{i=1}^{n} \mathrm{E}(\log f_{\delta_i}(Y_i;\theta_{\delta_i})|Y_i,\theta_*)$$

If we use conditioning argument,

$$\mathrm{P}(\delta_i = 1 |\, \mathrm{Y}_i = y_i, \theta_*) = \frac{\mathrm{P}(\delta_i = 1, \mathrm{Y}_i = y_i; \theta_*)}{P(\mathrm{Y}_i = y; \theta_*)} = w_{ih}(\theta_*) = \frac{p_h\, f_h(y_i,\ \theta_{h,*})}{\sum_{j=1}^{k} p_j f_j\,(y_i,\ \theta_{h,*})}$$

$w_h(\theta_*)$ is defined as membership weight of a data i to the cluster h.

Here, we can notice that

$$w_{ih} \geq 0, \quad \sum_{h=1}^{k} w_{ih} = 1$$

Thus,

$$Q(\theta_*, \theta) = \sum_{i=1}^{n} \sum_{j=1}^{k} w_{ih}(\theta_*) \cdot \log p_j f_j(\mathrm{Y}_i; \theta_j)$$

Below describes the EM-algorithm for clustering procedure.

1) Initialization: With this K-means clustering, the mean μ_i of each cluster S_i and covariance matrix can be calculated, and the weight w_j is randomly assigned.
2) E-step: Compute the expectation value of the weight. Here, p_i can be considered as weight function from the previous estimation.

   Then, we can calculate expected value of weight that data i belongs to the cluster (or Gaussian) h as below

   $$w_{ih}(\theta_*) = \frac{p_h\, f_h(y,\ \theta_{h,*})}{\sum_{j=1}^{k} p_j f_j\,(y,\ \theta_{h,*})}$$

   $$= \frac{w_h\, f_h(y_i,\ \theta_{h,*})}{w_1 f_1(y_i,\ \theta_{h,*}) + w_2 f_2(y_i,\ \theta_{h,*}) + w_3 f_3(y_i,\ \theta_{h,*}) + \cdots + w_k f_k(y_i,\ \theta_{h,*})}$$

   Again, $f_h(y,\ \theta_{h,*})$ is our density function,

   $$\prod_{j=1}^{k} f_{h,j}(y_j,\ \theta_{h,*}) = \prod_{j=1}^{k} \frac{1}{\sigma_j \sqrt{2\pi}} e^{-\frac{(y_i - \mu_j)^2}{2\sigma_j{}^2}}$$

Since our data, the term vector, is the vector rather than scalar, μ is a mean vector of a cluster, and a covarience matrix, Σ_j is used instead of variance σ_j.

3) M-step: Compute the mean and variance, or covariance matrix, from the estimated weight

$$\hat{p}_h = \frac{1}{n}\sum_{i=1}^{n} w_{ih}$$

$$\hat{\mu}_h = \frac{\sum_{i=1}^{n} w_{ih}\, y_i}{\sum_{i=1}^{n} w_{ih}}$$

$$\hat{\Sigma}_h = \frac{\sum_{i=1}^{n} w_{ih}\,(y_i - \hat{\mu}_h)(y_i - \hat{\mu}_h)^T}{\sum_{i=1}^{n} w_{ih}}$$

The E-step and M-step iteratively update the model and terminates the process when log-likelihood saturates. The threshold is set to *1e-06* in standard NLTK module for language model in Python. The EM-algorithm returns which term-vectors have a high chance to be in a cluster by clustering membership weight.

K-means and EM clustering both need the pre-determined number of clusters. Gap-statistics theory is applied in this study to expect the optimum number of clusters (Tibshirani et al, 2001). Gap-statistics measures compactness of clusters using the sum of within squares function (W_k), which is defined as

$$W_k = \sum_{r=1}^{k} \frac{1}{2|C_r|} D_r = \sum_{r=1}^{k} \frac{1}{2|S_r|} \sum_{i \in S_r} \|x_i - \mu\|^2,$$

where, D_r is defined as the sum of pairwise distances for all points in a cluster and S_r, x_i, and μ refer to the indices of observation in cluster r , i-th data point, and cluster mean, respectively. Gap statistics finds the value k, which maximizes the gap defined as

$$E * (log\ W_k) - log\ W_k,$$

where $E *$ denotes the expectation of a sample from the reference distribution, which is the null hypothesis of random noise.

Dimension selection in LSA

To use truncated singular value decomposition, it is necessary to decide the dimensions for the SVD. The diagonal matrix, Σ, orders the eigenvalues from largest to smallest. If there is a sudden drop in value between two eigenvalues as one moves along the diagonal, this fall can be interpreted as the transition of important eigenvalues to unimportant eigenvalues. Thus, for dimension reduction, it is common to search a sudden "elbow" of eigenvalues to decide a threshold. However, in the documents used here, eigenvalues tend to show an almost linear-type reduction. A plot of eigenvalue

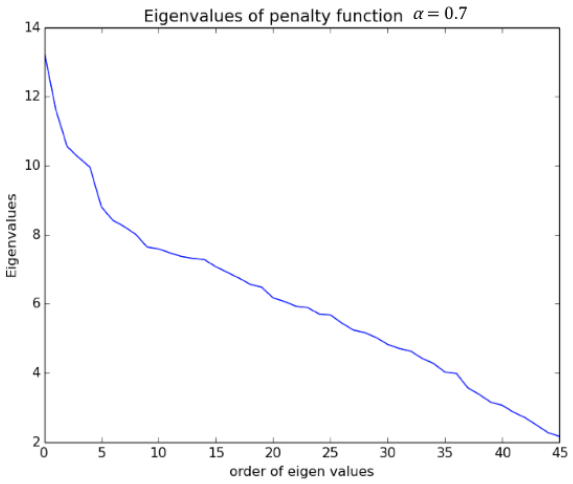


Figure 22 Example eigenvalues of diagonal matrix when penalty function α =0.7

value is shown in Figure 22. Thus, if a clear sudden drop in eigenvalues does not exist, we need another approach to find the threshold of eigenvalues for SVD.

In this research, we will test two common method to find the proper dimensions in SVD process (Jolliffe, 2002; Zhu & Ghodsi, 2006). These methods are introduced in the Principle Component Analysis (PCA) articles, however, since both PCA and LSA use SVD and try to find the right dimensionality in SVD, we can assume these methods can be also applied to our research.

1) Percent variance method

   If we let singular values in a diagonal matrix in LSA be $\{s_1, s_2, s_3, \ldots, s_n\}$, a percent variance method will keep the k eigenvalues which satisfies below equation.

$$\frac{\{s_1 + s_2 + s_3 + \cdots + s_k\}}{\sum_{i=1}^{n} s_i} \geq \xi$$

Typically, ξ is set to 70%, 80%, or 90%. The study of Wild et al.(Wild et al, 2005) found that for their work to automatically scoring essay using LSA, the ξ value 60% shows reasonably good result. In this study, we will test ξ values on a set of {10%, 20%,…, 90%}.

2) Profile likelihood method (Zhu & Ghodsi, 2006)

To find the proper dimensions to keep in SVD process, usually people tries to find the "elbow" or "sudden drop" in the eigenvalue profile. However, the eigenvalues in the research shows almost linearly decreasing profile, thus it is usually hard to find the "gap" in the profile, moreover, finding the breakage point from the profile (or screen plot) can be subjective (Zhu & Ghodsi, 2006). To avoid the shortcoming, this research adapts the study of Zhu and Ghods.

This method divides eigenvalues in two groups, $\mathcal{G}_1 = \{s_1 + s_2 + s_3 + \cdots + s_k\}$ and $\mathcal{G}_2 = \{s_k + s_{k+1} + s_{k+2} + \cdots + s_n\}$.

Then, the profile log-likelihood for k can be written as below,

$$\mathcal{L}_k(k) = \sum_{j=1}^{k} \log f_j(\mathrm{s}_j; \mu_1, \sigma) + \sum_{j=k+1}^{n} \log f_j(\mathrm{s}_j; \mu_2, \sigma)$$

where $f_j(\mathrm{s}; \mu_j, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\mathrm{s}-\mu_j)^2}{2\sigma^2}} \quad for\ j = 1, 2$

$$\mu_1 = \frac{\sum_{s_i \in \mathcal{G}_1} s_i}{k} \ , \qquad \mu_2 = \frac{\sum_{s_i \in \mathcal{G}_2} s_i}{n-k}$$

$$\sigma = \frac{(k-1)\sigma_1 + (n-k-1)\sigma_2}{n-2}$$

And σ_1 and σ_2 are the variances of $\mathcal{G}_1$and $\mathcal{G}_2$. Our objective is find the k value which satisfies the objective function $\underset{p=1,2,..,n}{\operatorname{argmax}} \mathcal{L}_k(p)$.

Algorithm

This section aims to illustrate how the actual algorithm works on a biological text to find morphologies and cluster biological texts according to the found morphologies. Pseudo code is in shown in APPENDIX D. Figure 23 shows an illustrative schematic of the entire process. Each computational step is discussed step by step below.

Step 1: Preprocessing the text and search paragraphs containing functional terms

Here, we consider the unit of a document a paragraph. We expand to a paragraph for passage sampling as the specific sentence containing the sought functional term may

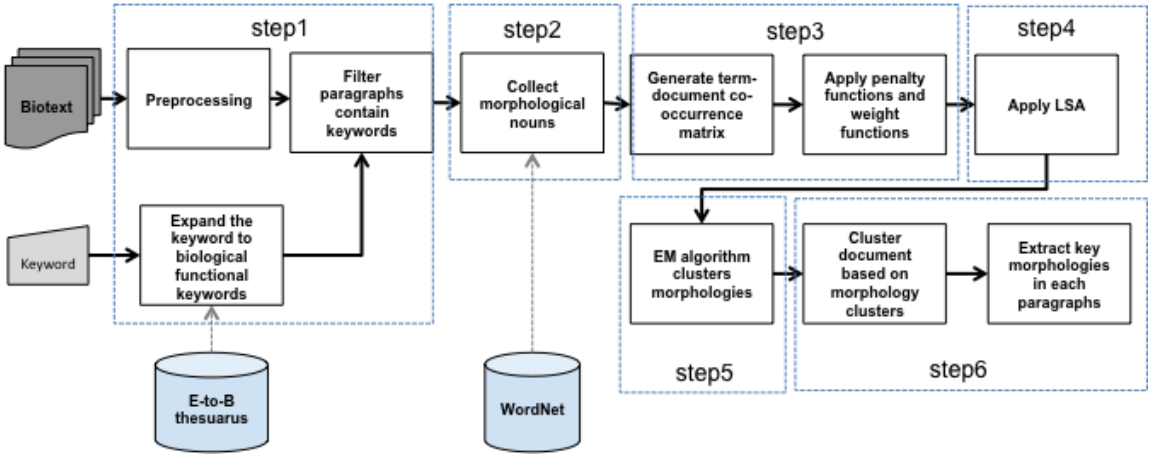


Figure 23 The computational process to the developed clustering algorithm

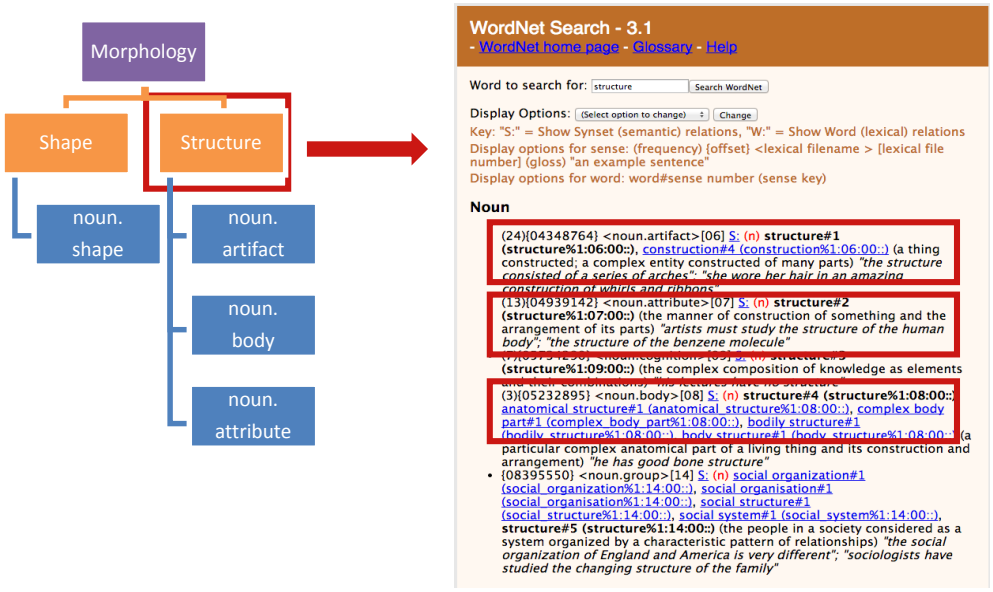


Figure 24 Decompose the definition of morphology: shape, and structure. Categories that can represent 'structure' are selected based on noun categories related to the definition of structure in WordNet

not have the associated description of the morphology. By searching the entire paragraph, we increase the chances of finding a clear morphological description. Thus, the first thing to do is divide the text file into paragraphs. The next preprocess is parsing a paragraph using the sentence parsing program, TreeTagger (Schmid, 1994). This parsing process finds a stem word (lemma) and tags a POS for every word in the paragraph.

The Engineering-to-Biology (E-to-B) thesaurus expands the input keyword to its corresponding biological functions. Then, the algorithm filters paragraphs that contain functional verbs. These keywords must be used in a verb form, which includes base

form, past tense, gerund, past participle, present singular, and present plural, based on TreeTagger POS tags.

Step 2: Collect morphological nouns in filtered paragraphs

As mentioned previously, the morphological nouns are identified using WordNet noun categories. The WordNet categories used to identify a discussion of morphology in the passage are attribute noun, artifact noun, body noun, and shape noun. These specific noun categories are highlighted in Figure 25. The reason for choosing artifact, attribute, body, and shape noun categories is an attempt to conform to the definition of morphology. As the definition of morphology indicates, morphology consists of two parts: shape and structure. Thus, the shape noun has been selected. The category of structural feature is chosen based on a close review of the definition of structure in WordNet. Figure 24 containes the definietion of structural features as used by WordNet.

Name	Description	Example
noun.Tops	unique beginner for nouns	Ex) entity, organism
noun.act	nouns denoting acts or actions	Ex) rush
noun.animal	nouns denoting animals	Ex) jaguar
noun.artifact	nouns denoting man-made objects	Ex) layer, stick
noun.attribute	nouns denoting attributes of people and objects	Ex) blue, stain, depth, trait, color
noun.body	nouns denoting body parts	Ex) tubule, teeth, hair
noun.cognition	nouns denoting cognitive processes and contents	Ex) theory, linguistic process, possibility
noun.communication	nouns denoting communicative processes and contents	Ex) music, speech
noun.event	nouns denoting natural events	Ex) happening, occurrence
noun.feeling	nouns denoting feelings and emotions	Ex) emotion, anger, joy
noun.food	nouns denoting foods and drinks	Ex) chocolate
noun.group	nouns denoting groupings of people or objects	Ex) law, police
noun.location	nouns denoting spatial position	Ex) Tokyo, origin, source, front
noun.motive	nouns denoting goals	Ex) conscience, moral sense
noun.object	nouns denoting natural objects (not man-made)	Ex) lake, rock, stone
noun.person	nouns denoting people	Ex) president, student
noun.phenomenon	nouns denoting natural phenomena	Ex) fog, tornado
noun.plant	nouns denoting plants	Ex) leaf, root
noun.possession	nouns denoting possession and transfer of possession	Ex) ownership
noun.process	nouns denoting natural processes	Ex) hydration, solavtion
noun.quantity	nouns denoting quantities and units of measure	Ex) inch, astronomical unit (au)
noun.relation	nouns denoting relations between people or things or ideas	Ex) function, hyponymy, connection
noun.shape	nouns denoting two and three dimensional shapes	Ex) helix, hexagon
noun.state	nouns denoting stable states of affairs	Ex) tension, dark
noun.substance	nouns denoting substances	Ex) water, urine, wood
noun.time	nouns denoting time and temporal relations	Ex) hour, minute

Figure 25 Noun categories in WordNet

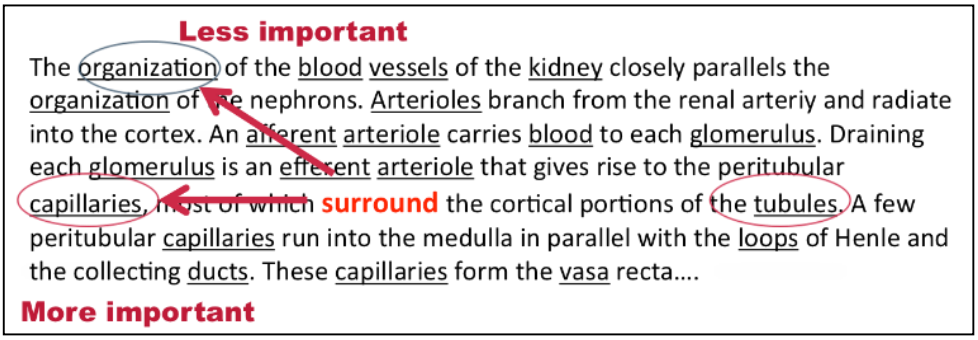


Figure 26 The importance of a morphology according to the distance of it from a functional verb

As we search for descriptions of morphology, we are interested both in 'a thing constructed' and 'the manner of construction of something' as taken from the original definition above. The artifact noun and attribute noun are thus added to the WordNet noun categories that are used to indicate a discussion of morphology in a passage. In biology, because of its distinction from man-made creation, natural components such as a tubule or tooth are usually expressed using body nouns instead of artifact nouns. Consequently, artifact noun, body noun, attribute noun, and shape noun are finally collected as morphology in text passages. To collect nouns only, the POS tag (as found in step 1) is used.

Some morphologies are expressed as a form of adjective. For example, "hexagonal" from "hexagonal shape" or "square" from "square pads" both indicate morphologies. However, the algorithm does not identify these terms as morphological nouns because they are tagged as adjectives. Thus, the lemma of an adjective is also examined through WordNet to determine if it is a morphological expression.

Step 3: Generate term-document matrix and apply weight functions

The rows of this matrix represent the morphological nouns as collected in Step 2, and the columns represent the paragraphs as filtered in Step 1. Each matrix entry indicates the frequency of its morphological noun (row) in its paragraph (column). After the co-occurrence matrix is generated, a penalty function is applied to the term-document matrix. The penalty function is to focus the algorithm on morphological nouns that are more likely to be related to the found functional term of interest. Note that the penalty function is different than the weight function. The weight function determines term importance in a passage, and in the entire collection of passages, based on term frequency. The penalty function operates locally in a passage and is focused more on finding morphological terms that are more likely to be describing the key functional verb.

The morphologies that are closely related to the function verb of interest are often located nearby in the passage, as shown in Figure 26. To increase the importance of the morphological nouns closer to the functional verb, this algorithm creates a penalty function and applies it to the term–document matrix. The penalty function, λ, is expressed as,

$$\lambda(i,j) = \alpha^{(\delta)} \times \mathrm{TF}(i,j),$$

where α and TF refer to a weight number less than 1, and term frequency, respectively. The symbol δ indicates the sentence order difference between the sentence that contains morphological noun and the sentence that contains the functional verb. For example, in

Figure 26, the morphology "organization" has a δ value of 3 and "tubule" has a zero δ value, since it is at the same sentence with a functional verb, "surround", containing sentence. Including penalty function, our new weight function is as in the below:

$$w(i,j) = \log(\lambda(i,j) + 1) \times log\frac{N}{df(i)} = \log\left(\alpha^{(\delta)} \times \mathrm{TF}(i,j) + 1\right) \times log\frac{N}{df(i)}.$$

This new weight function will be applied to a term-document matrix, S, before performing SVD.

Step 4: Apply LSA to morphology vector

The main objective of LSA is clustering similar morphological solutions. LSA can discover a latent relation between words by measuring frequency of co-occurring words in each document. For example, LSA can conclude that an "apple" and an "orange" are similar, even if they are not in the same document, by their shared words in

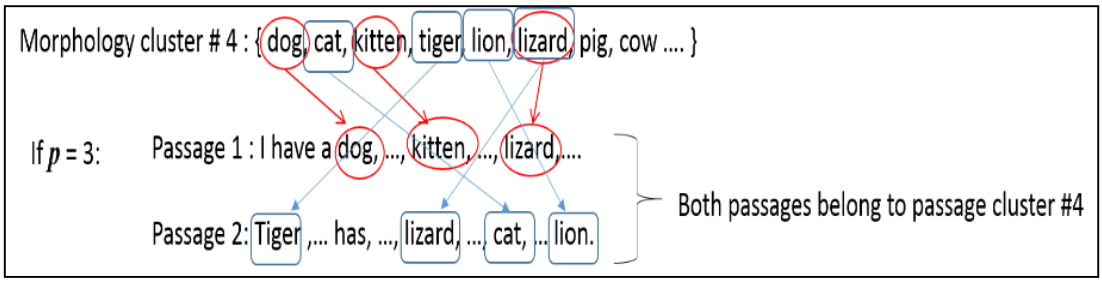


Figure 27 An example of passage clustering. If *p* =3, passages that have at least three morphologies from morphology cluster #4 will belong to the passage cluster #4. In this case, passage1 and passage 2 belong to passage cluster #4, because both passages have more than 3 morphologies from morphology cluster #4.

each document, such as “tree” or “juice.” Likewise, LSA can conclude that two morphological nouns, “pill” and “drug,” are similar even if they are not in the same biological passages.

After the dimensionality of LSA is decided as k, LSA maps the morphology vectors and document (paragraph) vectors to the k dimensional vector spaces($U_t \times \Sigma'$). By this process, the latent relation (dissimilarity) between morphological nouns can be discovered, and algorithm can prepare to apply a clustering algorithm to the truncated morphological vectors.

Step 5: Clustering morphological nouns by EM algorithm

Even though LSA is considered light clustering, it does not actually “group” words or documents, but provides the relativeness of word-to-word, document-to-document, and word-to-document. Therefore, this algorithm actually categorizes morphological nouns by applying a clustering process, the EM algorithm to the truncated term–Eigen matrix. The details of the EM algorithm used in this research was introduced previously.

Step 6: Cluster passages based on morphology groups

After morphologies are clustered from Step 5, passages are grouped according to morphology clusters. The passages that have more than ***p*** common morphologies from one morphology cluster are grouped together as a passage cluster.

Nomenclature

q: the number of morphological cluster

p: the minimum number of morphologies shared by passages in the same passage cluster

P_k: the set of k-th passage cluster

$n(P_k)$: the number of passages in k-th passage cluster

T: the summation of the number of passages in a passage cluster for all passage clusters,

$\sum_{k=1}^{n} n(P_k) = n(P_1) + n(P_2) + \cdots + n(P_n)$

A: the number of unique passages, $n(P_1 \cup P_2 \cup \ldots \cup P_n)$

Morphological cluster: a set of morphologies grouped together from Step 5

Morphology: an element in a morphological cluster – in other words, a morphological noun extracted by the system

Let us assume the algorithm has ***q*** morphology clusters. If passage 1 and passage 2 both have more than ***p*** morphologies from the 4th morphological cluster, they are grouped together as belonging to the 4th passage cluster (Figure 27). Consequently, if there is ***q*** number of morphology clusters, the number of passage clusters is also ***q***. This way of passage clustering indicates that one passage can have multiple morphological

solutions. In addition, a few passages are not included in any passage cluster and will be provided separately.

There is a several reasons why this algorithm does not cluster the document vector directly and chooses the grouping scheme presented in this subchapter. First, the algorithm wants to suggest morphological noun groups, and let designers select passages that contain a part of selected morphologies. Thus, the algorithm first clustered morphological vectors than document vectors. Second, one passage might has multiple morphological solutions. And we do not know how many morphological solution groups are contained in one passage. However, with this grouping scheme, we can assign multiple solution groups to one passage, even if we do not have the predetermined number of the morphological solution groups.

Deciding ***p*** involves considering the number of overlapped passages and the number of passages that are not clustered**.** If we let the summation of the number of passages in all passage clusters be ***T***, and the number of union be ***A***, then if ***p*** is too small, ***T*** might be too large, so an engineer would have to read too many overlapped passages. Conversely, if ***p*** is too large, the number of passages not included in any of the passage clusters will increase, which is not desirable. Thus, the number of ***p*** should be decided as that which makes ***T*** larger than ***A***, but minimizes the difference between ***T*** and ***A***. In other words, ***p*** is the result of compromising the number of overlapped passages and the number of passages remaining uncategorized. Usually, ***p*** has a value of 5 or 6 for a paragraph-unit document, but a user can decide the ***p*** value.

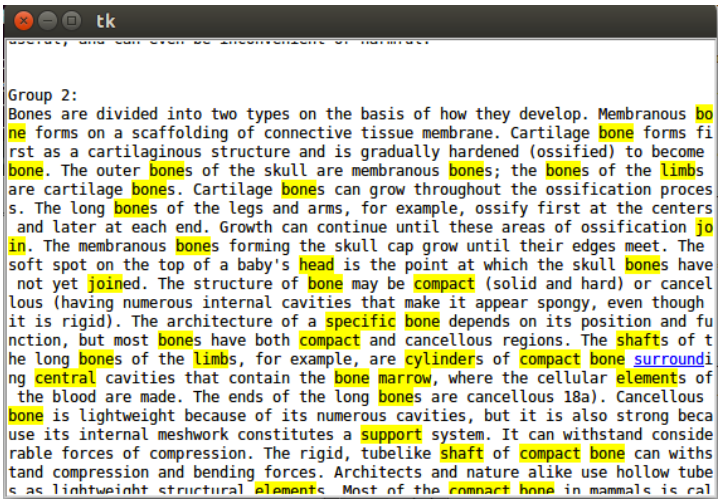


Figure 28 Sample screenshot of the algorithm

Finally, morphologies in each passage in a passage cluster are aligned according to their importance in a passage, using the truncated matrix S' generated by LSA. The value at the intersection of a column and a row in S' indicates the degree of relatedness of the row word in the column document. A value closer to 1 indicates that the term is more relevant to the passage. In this step, the term that has negative value in S' will be excluded, because the negative value indicates the poor relatedness of a document and a term. Finally, sorted morphologies in a passage are highlighted as in Figure 28.

Analysis of results

This section will examine how well the developed algorithm categorizes passages based on important morphologies. This research evaluated the algorithm by comparing morphologies extracted by the algorithm with morphologies manually

identified as important. The gold standard was made by two annotators. Morphologies that both annotators agreed important to conduct the functional verb in a passage are selected as the gold standard.

The algorithm is tested with the text corpus from the biological textbook, *Life: The science of biology* (Purves et al, 2003), and the input keyword of the engineering functional verb, "inhibit." As a result, 46 paragraphs have been retrieved and categorized according to morphological solutions. The minimum number of morphologies shared by passages in the same passage cluster, ***p***, is set to 5 to allow a uniform test condition. And system is set to retrieve top ten most important morphological nouns in a passage per one morphological cluster. Precision, recall and f-measure (Manning & Schütze, 1999) were used to evaluate the system, and can be defined as below.

$$Precision = \frac{Relevant\ retrieved\ document}{Retrieved\ document}$$

$$Recall\ = \frac{Relevant\ retrieved\ document}{Relavant\ document}$$

$$F1\ score = \frac{2 * (Precision \cdot Recall)}{Precision + Recall}$$

We tested three test cases:

1) Percent variance dimensions for 10%, 20%, …, 90%
2) Profile likelihood to select dimensions in SVD

Table 9 Precision for various percent variance dimensions and penalty ratio α

	α									
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
dim 0.1	0.1937	0.1776	0.1698	0.1495	0.1995	0.1965	0.1846	0.1763	0.1478	0.1625
dim 0.2	0.2033	0.1878	0.2143	0.1949	0.1941	0.1707	0.1626	0.1806	0.1786	0.1486
dim 0.3	0.1997	0.2118	0.1934	0.2213	0.2081	0.1938	0.1909	0.1671	0.1377	0.1517
dim 0.4	0.1852	0.1968	0.2104	0.2185	0.2217	0.1963	0.2457	0.1638	0.1690	0.1445
dim 0.5	0.1825	0.1786	0.1861	0.1894	0.2087	0.1948	0.1767	0.1647	0.1654	0.1760
dim 0.6	0.1558	0.1384	0.2281	0.2247	0.2137	0.1950	0.2076	***0.2626***	0.1663	0.1801
dim 0.7	0.2066	0.2019	0.2115	0.2181	0.2231	0.1951	0.1840	0.1635	0.2062	0.1595
dim 0.8	0.1360	0.1840	0.1965	0.2115	0.2076	0.1952	0.2028	0.1903	0.1923	0.1845
dim 0.9	0.2204	0.1455	0.2114	0.2231	0.2123	0.1724	0.1983	0.2163	0.2401	0.1835

Table 10 Recall for various percent variance dimensions and penalty ratio α

	α									
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
dim 0.1	0.4340	0.4387	0.4048	0.3323	0.4563	0.4481	0.4815	0.4596	0.4699	0.3590
dim 0.2	0.4260	0.4186	0.4928	0.4906	0.4664	0.4675	0.5042	0.4974	0.4259	0.4023
dim 0.3	0.3580	0.4473	0.3953	0.4253	0.5482	0.5383	0.4753	0.5653	0.4652	0.5068
dim 0.4	0.3192	0.3024	0.4310	0.5087	0.5192	0.5300	0.5694	0.5209	0.4887	0.4697
dim 0.5	0.2697	0.2557	0.3298	0.3596	0.4790	0.5408	0.4536	0.4520	0.5073	0.5186
dim 0.6	0.2399	0.2307	0.4076	0.4571	0.4596	0.4236	0.5008	**0.6026**	0.4805	0.5752
dim 0.7	0.3322	0.3711	0.3066	0.3617	0.5016	0.4793	0.5300	0.4442	0.5195	0.5198
dim 0.8	0.1947	0.3398	0.3056	0.4196	0.3762	0.5303	0.5319	0.5679	0.5751	0.4969
dim 0.9	0.3789	0.1888	0.3089	0.4653	0.4940	0.5655	***0.6256***	0.5970	0.5697	0.4576

Table 11 F1-score for various percent variance dimensions and penalty ratio α

	α									
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
dim 0.1	0.2679	0.2529	0.2393	0.2062	0.2776	0.2732	0.2669	0.2548	0.2248	0.2238
dim 0.2	0.2752	0.2592	0.2987	0.2790	0.2742	0.2500	0.2458	0.2650	0.2517	0.2170
dim 0.3	0.2564	0.2874	0.2598	0.2911	0.3017	0.2850	0.2724	0.2580	0.2125	0.2335
dim 0.4	0.2344	0.2384	0.2827	0.3057	0.3108	0.2864	0.3432	0.2492	0.2511	0.2210
dim 0.5	0.2177	0.2103	0.2379	0.2481	0.2907	0.2865	0.2543	0.2414	0.2494	0.2628
dim 0.6	0.1889	0.1730	0.2925	0.3013	0.2917	0.2670	0.2935	***0.3658***	0.2470	0.2743
dim 0.7	0.2547	0.2615	0.2503	0.2721	0.3089	0.2773	0.2731	0.2390	0.2952	0.2442
dim 0.8	0.1601	0.2387	0.2392	0.2812	0.2675	0.2853	0.2936	0.2850	0.2883	0.2691
dim 0.9	0.2787	0.1644	0.2510	0.3016	0.2970	0.2642	0.3012	0.3176	0.3378	0.2560

In addition, we explore the effectiveness of the α value in in the penalty function, whereby the algorithm is evaluated for α values from 0.1 to 1.0 in 0.1 increments. When α =1.0, it is same as no penalty function is applied . The number of common morphologies in a passage cluster ***p***, is set to 5 in this test. The results for test cases 1 is shown in Table 9, Table 10, and Table 11, and visualized in Figure 29, Figure 30, and Figure 31. The precision seems pretty low because our system select 10 morphological nouns from each passages per a morphological cluster. It means that if there are morphological solution types in one passage, the system can extract maximum 30 morphological nouns in a passage. Meanwhile, our average golden standards per one passage is 4.782 regardless to a morphological solution type, thus lower the precision values.

The first noticeable fact is that the result of the precision and f1-score, which resembles precision (Figure 29 and Figure 31), seems not really affected by dimensions.

That is because even though more correct answers are retrieved as the dimension increases, the total number of morphological nouns extracted by system increases, thus offsets the increase of correct answers. Similarly, it seems precision seems neither affected largely from the penalty rate alpha (α). One guess for this finding is that the amount of total retrieved solution might be too many, so that precision according to penalty rate is standardized downward.

However, the recall shows a trend according to the dimension reduction and the penalty rate. For most cases, as penalty rate increases, the recall gradually increases, but at certain point, it starts decreases. This means that there is an optimal penalty rate. In Table 9, Table 10, and Table 11, the red values are the best value for each dimension choices. As in Table 9 and Table 11, it seems the correlation of penalty rate and best value is weak. However, in Table 10, we can see that, for recall, the best recalls for each dimensions appear alpha between 0.6 and 0.9. Especially, the best recall appears in penalty rate 0.7 and percent variance at 90%, and second best appears in penalty rate 0.8 and percent variance at 60%. Also, in Figure 29, Figure 30, and Figure 31, we can see that the scores at $\alpha=1$ is less than the peak scores in most cases, and this indicates the if we can find the proper number of penalty rate, applying penalty function can give us better result than just applying TF-IDF. This is because that under the proper penalty function, weighting function reflects the notion that morphological nouns are those that appear close to the function keyword verb in the text.

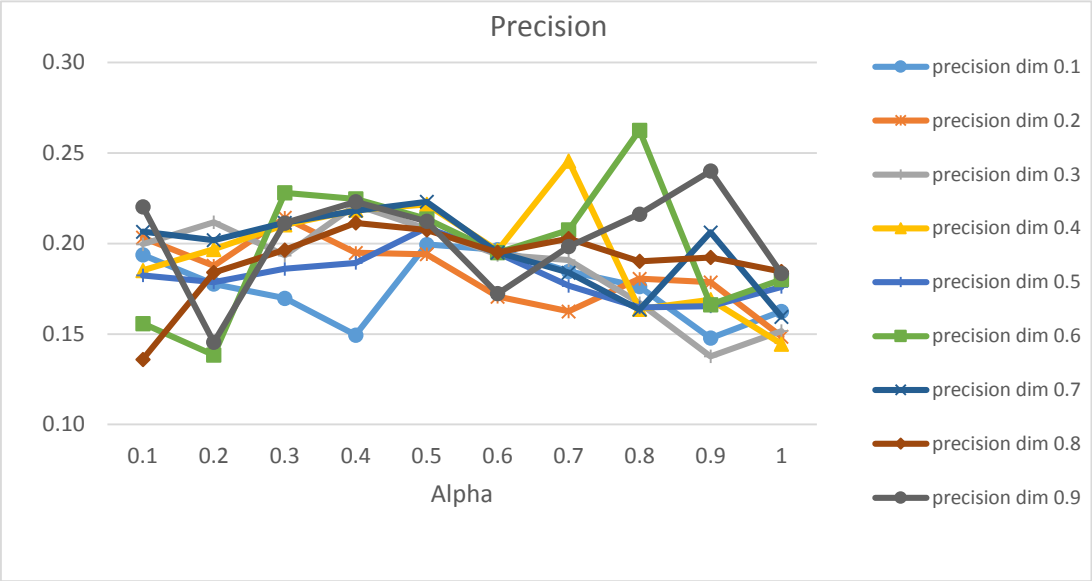


Figure 29 α – Precision graph

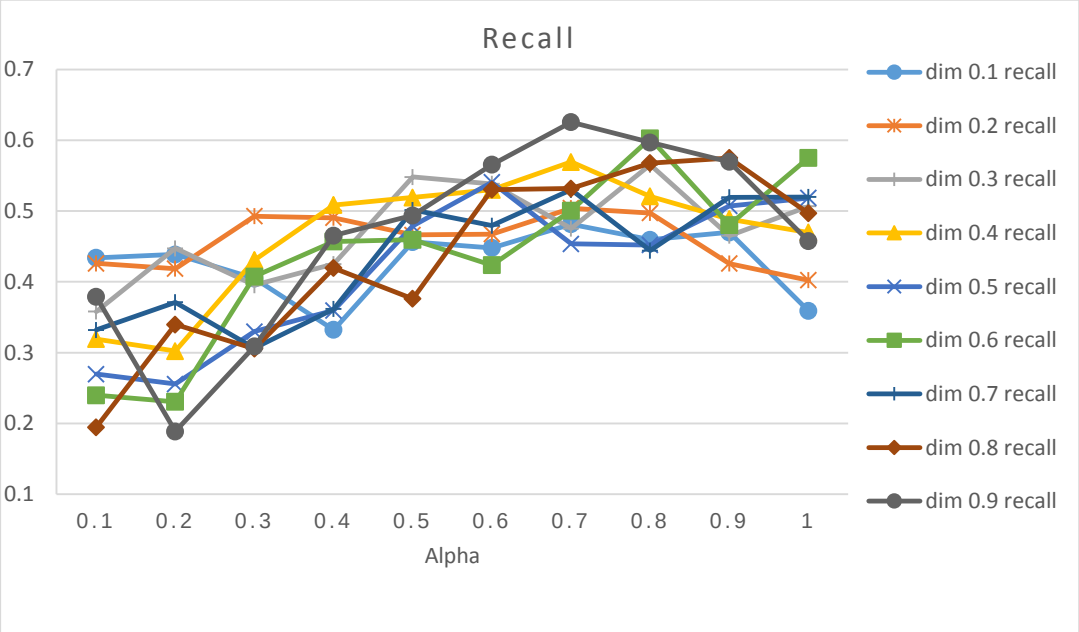


Figure 30 α – Recall graph

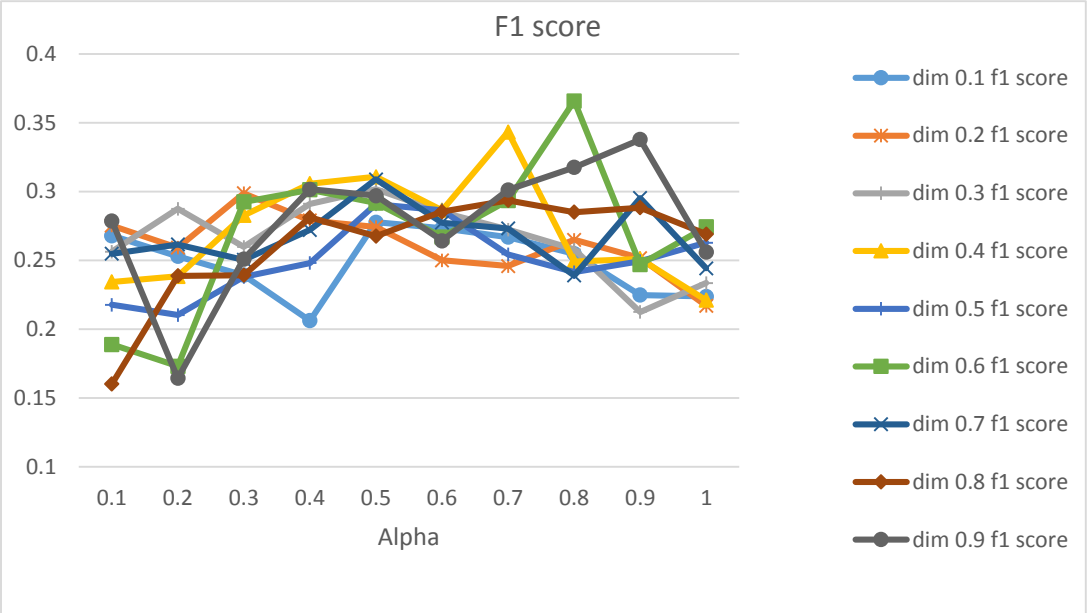


Figure 31 α – F1 score graph

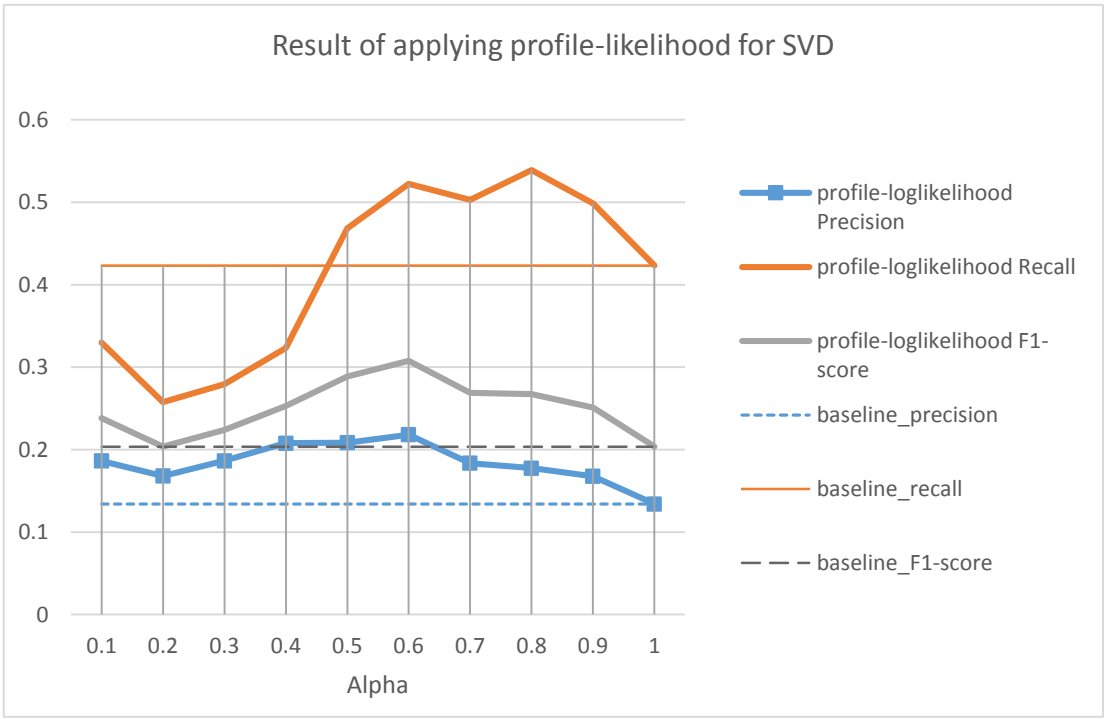


Figure 32 Result of applying profile-likelihood for SVD

The highlighted cells in Table 9, Table 10, and Table 11 are the values that shows better result than no-penalty function ($\alpha=1$). We can see that as dimensions selected in SVD increases, we need to increase α value as well to get a better result.

As we can see in the Figure 32, for precision and f1-score, it is always better to apply penalty function under profile-likelihood method, though both shows best result when the alpha value equals 0.6. However in recall, the low number of alpha value rather worsen the result, mostly because of its over-emphasis on the morphological nouns near the function verb. From the alpha value 0.5 to 0.9, the recall values show better results than the baseline, especially the alpha value 0.6, 0.7, 0.8, and 0.9. This agrees the findings in Percent variance dimensions test, which showed that best results regardless the dimension reduction appears in the range of alpha value from 0.6 to 0.9.

We can still see the limitation of the result. First, there is too much noises in the retrieved morphological nouns. This indicates that we have to control the maximum retrieved nouns per a passage, or we have to make a filter for morphological nouns extracted using WordNet. Also, we found several reasons, which diminish the recall value. First, the morphological nouns in golden standard sometimes not appeared in the WordNet. This is mainly because that WordNet cannot recognize the terminology that is rarely used in daily English. Second, some of golden standards are identified as a non-morphological term, such as signal. Because inter-raters showed tendency to select keywords that is important to conduct functional nouns in the system, rather than focus only on morphology. A lot of cases, a function can be achieved by combination of

various elements including signals, component or other flows. Thus, this can also viewed as a limitation of the morphology-based approach. However, still finding morphological information in biological passage is important, and we leave limitations of the current study to the future works.

Conceptual design example: Design an anti-impact fabric

This case study is designed to show how one would employ the developed algorithm in a conceptual design effort. The design objective of this case study is creating a new conceptual design for shock absorbing materials. In the simplified case illustrated here, we use a square pad. The following subsections provide details of the search algorithm outlined above and some additional design procedures.

Case study problem introduction

The objective of this case study is to create a conceptual design of a shock absorbing fabric. Specifically, we will employ to the search algorithm developed in this article and seek to identify a specific shock absorbing morphology used in nature that provides inspiration for an engineered shock absorbing fabric. This fabric might be applied to protective clothing as well as various other applications, such as floor mats or construction material. There is no specific restriction for dimensions or materials as the exercise is conceptual in nature. However, the goal is to identify a fabric that has impact or force dispersion advantages over conventional protective materials such as foam or fiber pad.

Procedures of the design

The case study follows the design steps using the developed keyword–morphology search tool. Design procedures are illustrated in Figure 33 to briefly illustrate both designer activities and computational activities. Briefly, the designer activities are as follow. 1) The designer draws a functional model for the design problem. 2) Next, the designer identifies a key function for the morphological search and enters the keyword in the developed algorithm. 3) After the algorithm clusters the biological text, the designer reviews the morphology represented in each cluster. 4) The designer selects the first passage in a cluster and reads it. If the selected passages are interesting to a designer, he or she continues to read and explore the specific morphology used by the natural systems to provide the needed functionality. 5) If passages in the cluster are not inspiring for the design problem, the designer skips the current passage cluster and moves on to another passage cluster related to the main function.

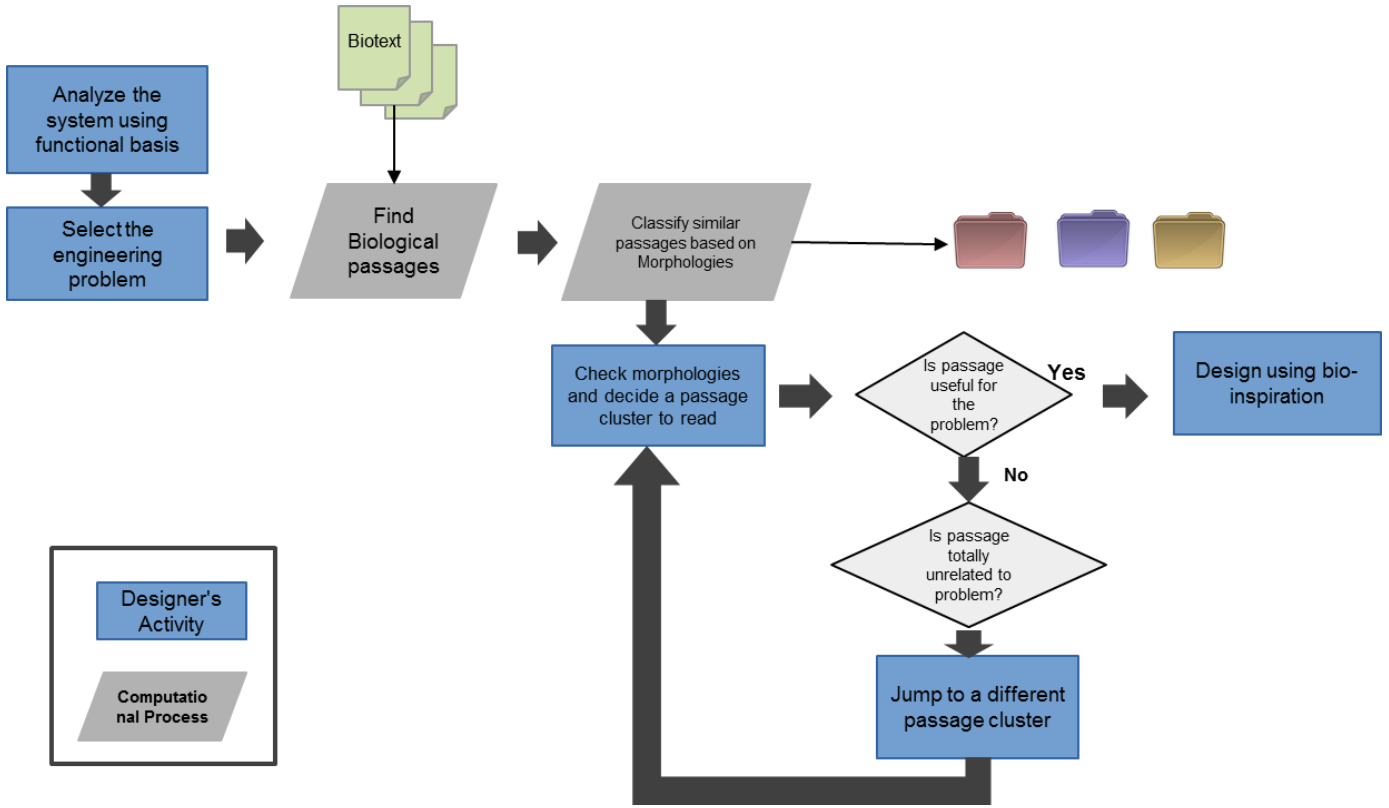


Figure 33 Designer's activity and computational procedures using the developed categorization algorithm

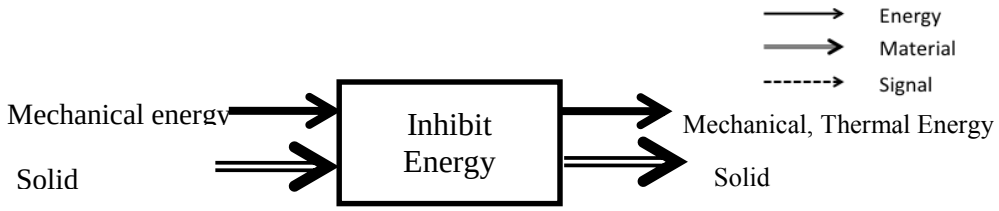


Figure 34 Black box model of an impact-free material

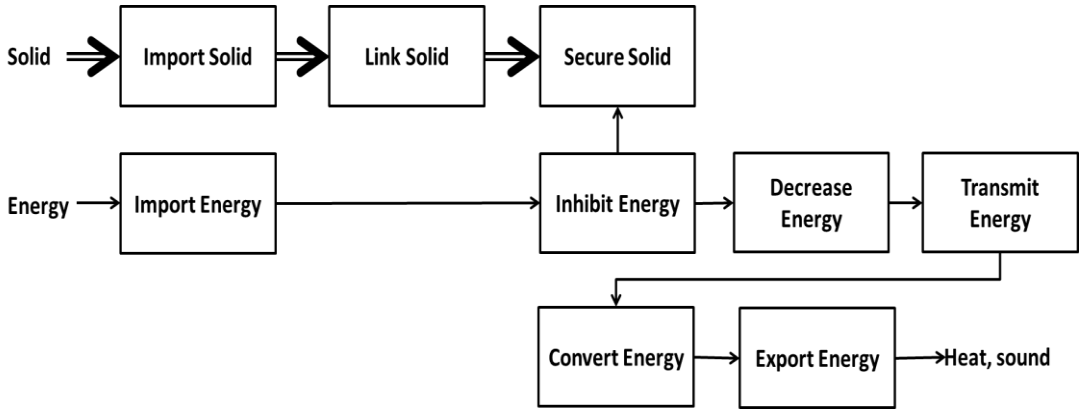


Figure 35 An example of functional model using the functional basis, expanding on the main function shown in Figure 34

For this case study, the functional model of an anti-impact fabric using the Functional Basis is shown in Figure 34 and Figure 35. This function keyword is input to the algorithm. In this case, we choose "inhibit" from the function "inhibit energy," which is the design function of interest for the shock absorbing material. Building on the E-to-B thesaurus, the algorithm will find all paragraphs in the search text that contain the keyword, "inhibit," and its biological correspondents, "cover," "destroy," "repress," and "surround".

Passage Cluster 1:

passage number 44: ['bone', 'central', 'compact', 'cylinder', 'element', 'force', 'head', 'join', 'limb', 'marrow', 'position', 'region', 'shaft', 'specific', 'support']

Passage Cluster 2:

passage number 13: ['addition', 'chain', 'duct', 'endocrine', 'exocrine', 'function', 'infection', 'lip', 'pancreas', 'process', 'release', 'secretion', 'short', 'stomach', 'time']

…

Passage Cluster 4:

passage number 9: ['cycle', 'follicle', 'gonadotropin', 'low', 'man', 'mechanism', 'pill', 'product', 'rate', 'source', 'use', 'uterus']

passage number 12: ['activity', 'cleft', 'drug', 'low', 'rate', 'use', 'work']

passage number 14: ['cycle', 'eye', 'man', 'product', 'rate', 'use']

passage number 17: ['ability', 'activity', 'benzodiazepine', 'drug', 'eye', 'man', 'pill', 'product', 'rate', 'relaxant', 'source', 'tranquilizer', 'use']

passage number 20: ['advantage', 'cycle', 'follicle', 'gonadotropin', 'low', 'man', 'mechanism', 'pill', 'rate', 'use', 'work']

passage number 30: ['cycle', 'low', 'man', 'rate']

…

Passage Cluster 7:

Figure 36 A part of output of the algorithm. Morphologies are ordered by its importance in a passage

After the algorithm collects the paragraphs that contain the key functional verbs, it filters morphological nouns in the text and categorizes collected morphological nouns and paragraphs. A part of the result of the algorithm is presented in Figure 36. The output presents the passage clusters and passages numbers included in each passage cluster. Morphologies in each passage are also provided next to the passage number.

After scanning a morphology list, a designer can be directly inspired from the morphologic words or decide to read a passage cluster based on morphologies.

The example passages from passage cluster #1 and passage cluster #4 are presented in Figure 37 and Figure 38, respectively. If one passage cluster contains few passages, this means that the morphological solutions in this passage cluster are shared by few passages. As a result, if an engineer wants relatively common morphological solutions, passage cluster#4 might be a good option. However, if an engineer wants rare or unique solutions, passage cluster #1 might be a better option than passage cluster #4.

After skimming the morphologies, the author was intrigued by passage cluster #1 and decided to read it (Figure 37). Reviewing the description in passage #44 of the morphology of bone, the designer was able to create some analogies between the morphology of bone and morphology for a shock absorbing material. Then, the author was inspired by highlighted keywords such as *compact bone*, *bone marrow*, *limb*, *shaft*, or *cylinder*. In the other way, the author was not very interested the morphologies contained in passage cluster #1 and skipped passages such as those in Figure 38. Moreover, despite not even reading the passages in passage cluster #7, the author was inspired by the morphologies such as *covering*, *membrane*, *mucus*, *layer*, or *plate* in the morphologies listed in the result.

After the author got enough inspirational words, a conceptual design sketch of the anti-impact fabric was generated as in Figure 39.

P44	Bones are divided into two types on the basis of how they develop. Membranous bone forms on a scaffolding of connective tissue membrane. Cartilage bone forms first as a cartilaginous structure and is gradually hardened (ossified) to become bone. The outer bones of the skull are membranous bones; the bones of the limbs are cartilage bones. Cartilage bones can grow throughout the ossification process. The long bones of the legs and arms, for example, ossify first at the centers and later at each end. Growth can continue until these areas of ossification join. The membranous bones forming the skull cap grow until their edges meet. The soft spot on the top of a baby's head is the point at which the skull bones have not yet joined. The structure of bone may be compact (solid and hard) or cancellous (having numerous internal cavities that make it appear spongy, even though it is rigid). The architecture of a specific bone depends on its position and function, but most bones have both compact and cancellous regions. The shafts of the long bones of the limbs, for example, are cylinders of compact bone **surround**ing central cavities that contain the bone marrow, where the cellular elements of the blood are made. The ends of the long bones are cancellous 18a). Cancellous bone is lightweight because of its numerous cavities, but it is also strong because its internal meshwork constitutes a support system. It can withstand considerable forces of compression. The rigid, tubelike shaft of compact bone can withstand compression and bending forces. Architects and nature alike use hollow tubes as lightweight structural elements. Most of the compact bone in mammals is called Haversian bone because it is composed of structural units called Haversian systems 18b). Each Haversian system is a set of thin, concentric bony cylinders, between which are the osteocytes in their lacunae. Through the center of each Haversian system runs a narrow canal containing blood vessels. Adjacent Haversian systems are separated by boundaries called glue lines. Haversian bone is resistant to fracturing because cracks tend to stop at glue lines.

Figure 37 A paragraph in passage cluster #1 among 46 paragraphs those have functional keyword ‘inhibit’. Only one paragraph is contained in passage cluster #1 because of its unique morphologies in the text. Again, Morphologies are highlighted and the functional keyword, surround, which is correspondent biological keyword to inhibit, is underlined.

P9	"Estrogen and especially progesterone secreted by the corpus luteum following ovulation are crucial to the continued growth and maintenance of the endometrium. In addition, these sex steroids exert negative feedback control on the pituitary, **inhibit**ing gonadotropin release and thus preventing new follicles from beginning to mature. If the egg is not fertilized, the corpus luteum degenerates on about day 26 of the cycle. Without the production of progesterone by the corpus luteum, the endometrium sloughs off, and menstruation occurs. The decrease in circulating steroids also releases the hypothalamus and pituitary from negative feedback control, so GnRH, FSH, and LH all increase. The increase in these hormones induces the next round of follicle development, and the ovarian cycle begins again. If the egg is fertilized, and a blastocyst arrives in the uterus and implants itself in the endometrium, a new hormone comes into play. A layer of cells covering the blastocyst begins to secrete human chorionic gonadotropin (hCG). This gonadotropin, a molecular homolog of LH, keeps the corpus luteum functional. Because hCG is present only in the blood of pregnant women, the presence of this hormone is the basis for pregnancy testing. These tissues derived from the blastocyst also begin to produce estrogen and progesterone, eventually replacing the corpus luteum as the most important source of these sexsteroids. Continued high levels of estrogen and progesterone prevent the pituitary from secreting gonadotropins; thus the ovarian cycle ceases for the duration of the pregnancy. The same mechanism is exploited by birth control pills, which contain synthetic hormones resembling estrogen and progesterone that prevent the ovarian cycle (but not the uterine cycle) by exerting negative feedback control on the hypothalamus and pituitary."
P17	"It binds to nicotinic ACh receptors, but does not activate them nearly as much as ACh does. Therefore, the skeletal muscles of an animal poisoned by curare cannot respond to motor neurons. The animal goes into flaccid (relaxed) paralysis and dies because its respiratory muscles stop contracting. Curare is used medically to treat severe muscle spasms and to prevent muscle contractions that would interfere with surgery. Another compound, atropine, which is extracted from the plant Atropa belladonna, binds to muscarinic ACh receptors and prevents ACh from activating them. Atropine is used medically to increase heart rate, decrease secretions of digestive juices, and decrease spasms of the gut. Eye doctors use atropine to dilate pupils for eye examinations. In the past, atropine was used cosmetically to make the eyes look big and dark-hence the plant species name belladonna, meaning """"beautiful lady."""" The ability of compounds extracted from plants and animals to bind to certain neurotransmitter receptors is the basis for neuropharmacology, the study and development of drugs that influence the nervous system. Natural products are still an important source of drugs, but today many drugs are designed and synthesized by chemists. A major group of drugs called benzodiazepines, for example, which are used as tranquilizers, muscle relaxants, and sleeping pills, are synthetic molecules that act on GABA receptors, open Cl- channels, hyperpolarize cells, and **inhibit** neural activity."

Figure 38 A part of example passages grouped by the system in passage cluster #4 among 46 paragraphs those have functional keyword 'inhibit'. Morphologies are highlighted and the functional keywords are underlined.

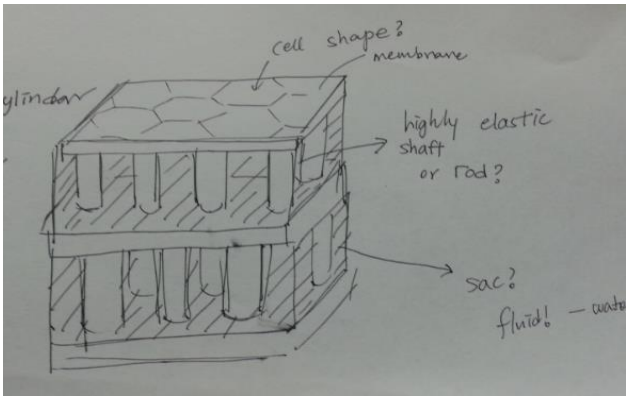


Figure 39 Sketch of a design inspired by morphological nouns found in the selected paragraphs in Figure 37

Redesigned anti-impact pad/fabric

The new design consists of four thin rubber layers, and two layers are paired. Distributed rubber cylinders are located between the thin layer pair. Air fills the cavity of hollow cylinder layers.

It is noteworthy that the bioinspired concept for a shock absorbing material presented here is similar to another bioinspired shock absorbing design. Figure 40 illustrates features from a shock absorber inspired from a woodpecker head, developed by Yoon and Park (Yoon & Park, 2011). The pad design in this case study and Figure 40 have a few similarities. Both concepts use multiple layers, cavities, and damping elastic materials. The design illustrated in Figure 40 is a more fleshed out concept. The shock absorbing system illustrated in Figure 17 is based on inspiration from a system with

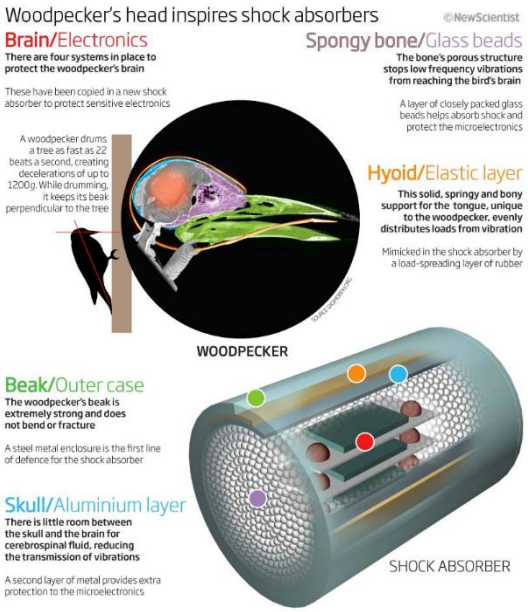


Figure 40 Woodpecker head inspired anti-impact design developed by scholars of university of california, berkeley (Marks, 2011; Yoon & Park, 2011)

which the designers had familiarity. Clearly, the woodpecker is a curiosity as it can bang its head into wood at high velocities, almost 20 times per a second, and suffer no damage. The keyword and morphological search method here found a similar system for inspiration. Of importance, the designer needed no familiarity with biological systems that provided shock absorption. The inspirational systems for a shock absorbing material are found by the algorithm presented here. As such, the keyword morphological search allows designers with limited biological knowledge search nature for solutions to design problems.

In this illustrative conceptual design example, we explored the application of the keyword morphological search applied to the design of shock absorbing material. The

tool finds biological morphologies used to solve the function “inhibit” in nature and clusters them based on similar passages. This clustering allows engineers to review morphological strategies in a more organized and focused manner than if the passages were returned in a random, or flat, manner. Based on the clustered passages, the designer can review one passage and conclude that no further passages in that cluster are worth reviewing. Finally, the case study design was compared to the existing bioinspired shock absorbing design.

CHAPTER VI

CONCLUSION AND FUTURE WORK

This dissertation provides a method that improves text based concept generation for bioinspired design. In detail, this research effort serves to remove the lexical gap between biologists and engineers that exists in biological text by importing lexical substitution theories. The first attempt to provide a solution to this lexical gap problem was developing the substitution algorithm using the well-known lexical database, WordNet. In this first algorithm, the essential feature was processing a word with multiple meanings using clustering WSD. Briefly, the algorithm's purpose was to create a term-to-term matrix, which consisted of candidate words excerpted from WordNet in rows and context words around target words in columns. Then, K-means clustering disambiguated the similar words to a target word based on the co-occurrence information of candidate words and context words information in the matrix. This algorithm reaches the 46.9% mode recall value, which outperforms the two baselines. Aside from developing the substitution algorithm, WordNet's coverage of biological words has been evaluated. WordNet contains few biological terms, about 40%, and can be considered not suitable for processing biological terms.

To overcome the limitation of WordNet, this dissertation newly adapted other lexical sources – ITIS, Wikipedia and WordNik. Using these sources, the coverage of biological terms raised from 40% to 86%, which can be seen as significant increase. The main assumption for using these new lexical sources is due to the rareness of biological

terms. Words omitted in WordNet are usually one meaning words because of their rareness and, consequently, we do not have to consider the WSD problem. To explain the details of these new lexical sources, ITIS is used to substitute scientific names and abbreviated scientific names. Using taxonomic hierarchy in ITIS, the developed algorithm substitutes the target word by a broader hierarchical name or a commonly used name of the scientific name. In addition, Wikipedia is used to process bioterms that are not included in WordNet and ITIS. The developed algorithm transfers a target word with a corresponding noun phrase in a Wikipedia article by heuristically collected identifiers.

Finally, WordNik is used to make up Wikipedia. Specifically, WorkdNik processes many adjectives and adverbs not contained in Wikipedia because of its encyclopedic characteristics. The combination of these lexical sources in the algorithm has been evaluated. We found that 63% of substitutions performed by the algorithm was agreed upon by two annotators, which can be considered acceptable results.

Additionally, this dissertation focuses on effectively delivering the biological text and the design solutions contained within the text. This objective is achieved using data mining to find morphological solutions within biotext and categorizing the text based on the solutions found. For the categorization, the widely applied text categorizing technique known as LSA was adapted. Morphology has been clarified in this research as a comprehensive expression of a structure and a shape of a biological or engineering system. Specifically, nouns about the body, artifacts, attributes, and shapes are considered morphological nouns. These morphological nouns could be found by WordNet's noun categories, which are manually tagged by linguists. Before categorizing

using LSA, the textual distance between biological functions was considered. A penalty function has also been developed and presented in this dissertation. Even though the precision and recall is not high, it has been proven that morphological extraction using a penalty function better follows the human extraction of morphological solutions than the baseline, which only uses LSA techniques. Finally, the biological text could be categorized after the morphological noun categorization was performed.

Contribution

This dissertation has made several contributions as listed below.

1. This research is the first attempt to translate biological terminologies for the concept generation phase in bioinspired design. Many other researches tried to build a thesaurus, which connects bioterms to corresponding engineering terms. However, these methodologies are limited to specific words, especially engineering functional terms. This research aims to substitute every bioterm that cannot be understandable to college level engineers in a given biotext and to enhance accessibility of biotext to engineers.
2. Secondly, this research developed several new methods not developed by conventional lexical substitution theories. The approach for biological terms should be different from daily lexicons due to the domain specific characteristics of biotext. Biological scientific names could be substituted using taxonomic hierarchy in this research. Since scientific names are mostly unrecognizable to engineers, the broader name of the specific scientific name might increase the accessibility of biotext in the concept generation process. Also, unlike other lexical substitution theories that require

the collection of candidate words, this research aimed to substitute a target word using an open source lexical database without candidate word collection. This approach may be applied to any field other than biology.

3. This research improved natural language processing of bioinspired design by extracting morphological solutions. Since morphology (precisely, structure) is an important factor in the design process, this research can be used to enlarge the solution space of engineering problems without wasting the tremendous efforts of engineers.

Future work

Future work remains for this research as listed below.

1. Since clustering based WSD using WordNet is not accurate, improved WSD processes are required to substitute target words. Adapting web-based candidate collections or web-based WSD processes may be more beneficial considering these methods have shown increased accuracy for the WSD problem (McCarthy & Navigli, 2007).

2. Acronyms of biological terminology, such as protein acronyms, are not considered in this research. WSD processing of this acronym is still an open question and widely studied in bioinformatics. Thus, deeper research regarding the biological acronym WSD is required.

3. Even though we assume that words not in WordNet are considered monosemic words, some words in Wikipedia still have a WSD problem.

4. The morphological extraction process produces a large noise, which is limitation of this initial research effort. This noise is likely because we did not consider the WSD problem of morphological nouns in biotext. Usually, one noun can have multiple noun categories, and the algorithm collects nouns if it has at least one morphological noun category among all categories. To solve this problem, morphological nouns should be disambiguated to its exact use in context.

REFERENCES

Adafre, S. F. & De Rijke, M. (2006) Finding similar sentences across multiple languages in wikipedia, *Proceedings of the 11th Conference of The European Chapter of the Association for Computational Linguistics*. Trento, Italy.

Agirre, E. & Rigau, G. (1996) Word sense disambiguation using conceptual density, *Proceedings of the 16th Conference on Computational Linguistics*. Copenhagen, Denmark.

Al′tshuller, G. S. (1999) *The innovation algorithm: TRIZ, systematic innovation and technical creativity*. Worcester, MA, USA: Technical Innovation Center, Inc.

Ananiadou, S. & McNaught, J. (2006) *Text mining for biology and biomedicine*. Boston, MA, USA: Artech House.

Arnold, C. R., Stone, R. B. & McAdams, D. A. (2008) MEMIC: An interactive morphological matrix tool for automated concept generation, *Proceedings of the 2008 Industrial Engineering Research Conference*. Vancouver, British Columbia, Canada.

Banerjee, S. & Pedersen, T. (2002) An adapted lesk algorithm for word sense disambiguation using wordnet, *Computational Linguistics and Intelligent Text Processing*. Mexico City, Mexico.

Benami, O. & Jin, Y. (2002) Creative stimulation in conceptual design, *ASME 2002 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Montreal, Quebec, Canada.

Benyus, J. M. (1997) *Biomimicry*. New York, NY, USA: William Morrow

Bilmes, J. A. (1998) A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. *International Computer Science Institute*, 4(510), 126.

Bingham, E. & Mannila, H. (2001) Random projection in dimensionality reduction: Applications to image and text data, *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Francisco, CA, USA.

Bird, S., Klein, E. & Loper, E. (2009) *Natural language processing with python*. Sebastopol, CA, USA: O'Reilly Media, Inc.

Bohm, M. R., Vucovich, J. P. & Stone, R. B. (2005) Capturing creativity: Using a design repository to drive concept innovation, *ASME 2005 International Design Engineering Technical Conferences and Computers And Information in Engineering Conference*. Long Beach, CA, USA.

Bonnardel, N. (2000) Towards understanding and supporting creativity in design: Analogies in a constrained cognitive environment. *Knowledge-Based Systems*, 13(7-8), 505-513.

Bryant, C. R., McAdams, D. A., Stone, R. B., Kurtoglu, T. & Campbell, M. I. (2005) A computational technique for concept generation, *ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Long Beach, CA, USA.

Calogero, G. & Di Marco, G. (2008) Red sicilian orange and purple eggplant fruits as natural sensitizers for dye-sensitized solar cells. *Solar Energy Materials and Solar Cells*, 92(11), 1341-1346.

Chakrabarti, A., Sarkar, P., Leelavathamma, B. & Nataraju, B. S. (2005) A functional representation for aiding biomimetic and artificial inspiration of new ideas. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 19(2), 113-132.

Cheong, H., Chiu, I., Shu, L. H., Stone, R. B. & McAdams, D. A. (2011) Biologically meaningful keywords for terms of the functional basis. *Journal of Mechanical Design*, 132(2), 1-11.

Cheong, H., Shu, L. H., Stone, R. B. & McAdams, D. A. (2008) Translating terms of the functional basis into biologically meaningful keywords, *ASME 2008 International Design Engineering Technical Conferences of Computers and Information in Engineering Conference*. New York, NY, USA.

Choueka, Y. & Lusignan, S. (1985) Disambiguation by short contexts. *Computers and the Humanities*, 19(3), 147-157.

Chujo, K. & Utiyama, M. (2006) Selecting level-specific specialized vocabulary using statistical measures. *System*, 34(2), 255-269.

Coxhead, A. (2000) A new academic word list. *Tesol Quarterly*, 34(2), 213-238.

Dempster, A. P., Laird, N. M. & Rubin, D. B. (1977) Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1), 1-38.

Design Engineering Lab *Bio search* Available online: http://www.designengineeringlab.org/delabsite/BioSearch.html [Accessed 2012.10.01].

Dieter, G. E. (1991) *Engineering design: A materials and processing approach*. New York, NY, USA: McGraw-Hill.

Erden, M. S., Komoto, H., van Beek, T. J., D'Amelio, V., Echavarria, E. & Tomiyama, T. (2008) A review of function modeling: Approaches and applications. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 22(2), 147-169.

Fellbaum, C. (2010) WordNet. *Theory and Applications of Ontology: Computer Applications*, 231-243.

Gabrilovich, E. & Markovitch, S. (2007) Computing semantic relatedness using Wikipedia-based explicit semantic analysis, *International Joint Conference on Artificial Intelligence*. Hyderabad, India.

Geertzen, J. (2012) *Inter-rater agreement with multiple raters and variables*, 2012. Available online: https://mlnl.net/jg/software/ira/ [Accessed 2015.07.01].

Gero, J. S. & Kazakov, V. (1999) Adapting evolutionary computing for exploration in creative designing. *Computational Models of Creative Design* 4, 175-186.

Giuliano, C., Gliozzo, A. & Strapparava, C. (2007) FBK-irst: Lexical substitution task exploiting domain and syntagmatic coherence, *Proceedings of the 4th International Workshop on Semantic Evaluations*. Prague, Czech Republic.

Glier, M. W., McAdams, D. A. & Linsey, J. S. (2013) An experimental investigation of analogy formation using the engineering-to-biology thesaurus, *ASME 2013 International Design Engineering Technical Conferences and Computers and Information Engineering Conference*. Portland, Oregon, USA, August 4-7.

Globerson, A. & Tishby, N. (2003) Sufficient dimensionality reduction. *The Journal of Machine Learning Research*, 3, 1307-1331.

Hacco, E. & Shu, L. H. (2002) Biomimetic concept generation applied to design for remanufacture, *ASME 2002 Design Engineering Technical Conference*. Montreal, Canada.

Hassan, S., Csomai, A., Banea, C., Sinha, R. & Mihalcea, R. (2007) UNT: Subfinder: Combining knowledge sources for automatic lexical substitution, *Proceedings of the 4th International Workshop on Semantic Evaluations*. Prague, Czech Republic.

Heatley, A., Nation, P. & Coxhead, A. (1994) *Range*, 1994. Available online: http://www.victoria.ac.nz/lals/about/staff/paul-nation [Accessed 2015.07.09].

Hirtz, J., Stone, R. B., McAdams, D. A., Szykman, S. & Wood, K. L. (2002) A functional basis for engineering design: Reconciling and evolving previous efforts. *Research in Engineering Design-Theory Applications and Concurrent Engineering*, 13(2), 65-82.

Integrated Taxonomic Information System *Integrated taxonomic information system (ITIS)* Available online: www.itis.gov [Accessed 2015.05.01].

Jackson, S. R., Newport, R., Mort, D. & Husain, M. (2005) Where the eye looks, the hand follows: Limb-dependent magnetic misreaching in optic ataxia. *Current Biology*, 15(1), 42-46.

Jolliffe, I. (2002) *Principal component analysis*. Hoboken, NJ, USA: John Wiley & Sons, Ltd.

Krallinger, M., Erhardt, R. A.-A. & Valencia, A. (2005) Text-mining approaches in molecular biology and biomedicine. *Drug Discovery Today*, 10(6), 439-445.

Laham, T. K. L. D. & Foltz, P. (1998) Learning human-like knowledge by singular value decomposition: A progress report, *Advances in Neural Information Processing Systems 10: Proceedings of the 1997 Conference*. Denver, Colorado, USA.

Landauer, T. K., Foltz, P. W. & Laham, D. (1998) An introduction to latent semantic analysis. *Discourse Processes*, 25(2-3), 259-284.

Landis, J. R. & Koch, G. G. (1977) The measurement of observer agreement for categorical data. *Biometrics*, 33(1), 159-174.

Manning, C. D., Raghavan, P. & Schütze, H. (2008) *Introduction to information retrieval*, *1*. Cambridge, United Kingdom: Cambridge University Press

Manning, C. D. & Schütze, H. (1999) *Foundations of statistical natural language processing*. Cambridge, MA, USA: MIT press.

Marcus, M. P., Marcinkiewicz, M. A. & Santorini, B. (1993) Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19(2), 313-330.

Marks, P. (2011) Woodpecker inspires shock absorbers. *New Scientist*, 209(2798), 21-21.

Martinez, D., Kim, S. N. & Baldwin, T. (2007) MELB-MKB: Lexical substitution system based on relatives in context, *Proceedings of the 4th International Workshop on Semantic Evaluations*. Prague, Czech Republic.

McCarthy, D. (2002) Lexical substitution as a task for WSD evaluation, *Proceedings of the ACL-02 Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*. Philadelphia, PA, USA.

McCarthy, D. & Navigli, R. (2007) SemEval-2007 task 10: English lexical substitution task, *Proceedings of the 4th International Workshop on Semantic Evaluations*. Prague, Czech Republic.

McCarthy, D. & Navigli, R. (2009) The english lexical substitution task. *Language Resources and Evaluation*, 43(2), 139-159.

Miles, L. (1961) *Techniques of value analysis and engineering*. New York, NY, USA: McGraw-Hill.

Milne, D. & Witten, I. H. (2008) Learning to link with wikipedia, *Proceedings of the 17th ACM Conference on Information and Knowledge Management*. Napa Valley, CA, USA, 1458150.

Mogilner, A. & Keren, K. (2009) The shape of motile cells. *Current Biology*, 19(17), 762-771.

Nagel, J. K. S. & Stone, R. B. (2010) A computational concept generation technique for biologically-inspired, engineering design, *Design Computing And Cognition DCC'10*. Stuttgart, Germany.

Nagel, J. K. S. & Stone, R. B. (2011) A systematic approach to biologically-inspired engineering design, *ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Washington, D.C., USA.

Nagel, J. K. S., Stone, R. B. & McAdams, D. A. (2010) An engineering-to-biology thesaurus for engineering design, *ASME 2010 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Montreal, Quebec, Canada.

Nakov, P., Popova, A. & Mateev, P. (2001) Weight functions impact on LSA performance, *EuroConference RANLP*. Tzigov Chark, Bulgaria.

Olie, R. A., Durrieu, F., Cornillon, S., Loughran, G., Gross, J., Earnshaw, W. C. & Golstein, P. (1998) Apparent caspase independence of programmed cell death in *dictyostelium*. *Current Biology*, 8(17), 955-S1.

Otto, K. & Wood, K. L. (2001) *Product design: Techniques in reverse engineering, systematic design, and new product development*. New York, NY, USA Prentice-Hall.

Pahl, G., Beitz, W., Feldhusen, J. & Grote, K.-H. (2007) *Engineering design: A systematic approach, 157*. London, United Kingdom: Springer.

Parker, A. R. & Townley, H. E. (2007) Biomimetics of photonic nanostructures. *Nature Nanotechnology*, 2(6), 347-353.

Purves, W. K., Orians, G. H., Sadava, D. & Heller, H. C. (2003) *Life: The science of biology: Volume iii: Plants and animals, 3*. London, United Kingdom: Macmillan.

Qian, L. & Gero, J. S. (1992) A design support system using analogy, *Artificial Intelligence In Design'92*. London, United Kingdom: Springer, 795-813.

Queller, D. C. & Strassmann, J. E. (2003) Eusociality. *Current Biology*, 13(22), R861-R863.

Salton, G. & Buckley, C. (1988) Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5), 513-523.

Sarkar, P., Phaneendra, S. & Chakrabarti, A. (2008) Developing engineering products using inspiration from nature. *Journal of Computing and Information Science in Engineering*, 8(3), 1-9.

Schmid, H. (1994) Treetagger. *TC project at the Institute for Computational Linguistics of the University of Stuttgart.* University of Stuttgart.

Schutze, H. (1998) Automatic word sense discrimination. *Computational Linguistics*, 24(1), 97-123.

Shu, L. H., Ueda, K., Chiu, I. & Cheong, H. (2011) Biologically inspired design. *CIRP Annals - Manufacturing Technology*, 60(2), 673-693.

Sinha, R. & Mihalcea, R. (2014) Explorations in lexical sample and all-words lexical substitution. *Natural Language Engineering*, 20(1), 99-129.

Sleator, D. D. K. & Temperley, D. (1991) *Parsing english with a link grammar*.Pittsburgh, PA, USA.

Spyns, P. (1996) Natural language processing. *Methods of Information in Medicine*, 35(4), 285-301.

Srinivasan, V. & Chakrabarti, A. (2009) SAPPhIRE - an approach to analysis and synthesis, *Proceedings of ICED 09, the 17th International Conference on Engineering Design*. Palo Alto, CA, USA.

Stone, R. B. & Wood, K. L. (2000) Development of a functional basis for design. *Journal of Mechanical Design*, 122(4), 359-370.

Stroble, J. K., Stone, R. B., McAdams, D. A., Goeke, M. S. & Watkins, S. E. (2009) Automated retrieval of non-engineering domain solutions to engineering problems, *Proceedings of the 19th CIRP Design Conference - Competitive Design*. Cranfield, United Kingdom.

Suryavanshi, S., Edde, B., Fox, L. A., Guerrero, S., Hard, R., Hennessey, T., Kabi, A., Malison, D., Pennock, D. & Sale, W. S. (2010) Tubulin glutamylation regulates ciliary motility by altering inner dynein arm activity. *Current Biology*, 20(5), 435-440.

Tibshirani, R., Walther, G. & Hastie, T. (2001) Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2), 411-423.

Ullman, D. (2009) *The mechanical design process*. New York, NY, USA: McGraw-Hill

United States Census 2000 *Frequently occurring surnames from census 2000* Available online: http://www.census.gov/main/www/cen2000.html [Accessed 2015.07.09].

Vakili, V. & Shu, L. H. (2001) Towards biomimetic concept generation, *Proceedings of the ASME Design Engineering Technical Conference*. Pittsburgh, PA, USA.

Vattam, S., Wiltgen, B., Helms, M., Goel, A. K. & Yen, J. (2011) DANE: Fostering creativity in and through biologically inspired design, in Taura, *Design Creativity 2010*. London, United Kingdom: Springer. 115-122.

Villalon, J. & Calvo, R. A. (2009) Single document semantic spaces, *Proceedings of the 8th Australasian Data Mining Conference*. Melbourne, Australia.

West, M. P. (1953) *A general service list of english words: With semantic frequencies and a supplementary word-list for the writing of popular science and technology*. London, United Kingdom: Longman, Green and Co.

Wikipedia *Wikipedia* Available online: http://www.wikipedia.org/ [Accessed 2015 01.01].

Wild, F., Stahl, C., Stermsek, G. & Neumann, G. (2005) Parameters driving effectiveness of automated essay scoring with LSA, *Proceedings of the 9th CAA Conference*. Loughborough, United Kingdom.

Yoon, S. H. & Park, S. (2011) A mechanical analysis of woodpecker drumming and its application to shock-absorbing systems. *Bioinspiration and Biomimetics,* 6(1), 016003+12.

Yuret, D. (2007) KU: Word sense disambiguation by substitution, *Proceedings of the 4th International Workshop on Semantic Evaluations*. Prague, Czech Republic.

Zhao, S., Zhao, L., Zhang, Y., Liu, T. & Li, S. (2007) HIT: Web based scoring method for english lexical substitution, *Proceedings of the 4th International Workshop on Semantic Evaluations*. Prague, Czech Republic.

Zhu, M. & Ghodsi, A. (2006) Automatic dimensionality selection from the scree plot via the use of profile likelihood. *Computational Statistics and Data Analysis*, 51(2), 918-930.

Zwicky, F. (1969) *Discovery, invention, research through the morphological approach*. New York, NY, USA: Macmillan.

APPENDIX A

PSEUDO CODE OF THE DEVELOPED ALGORITHM FOR CHAPTER III

```
Input: Engineering function E
Output: Translated biotext


keyword == raw_input(E)
Find Biological functional term B from keyword
for sentence in Biocorpus:
        if B in sentence:
                save sentence
read sentence
for word in sentence:
    if word not in BNC_8000word_list & stopword_list:
        translation = Translation(word)
        word = translation



def Translation(word):
    if word in WordNet:
        find synonyms
        sense = count(synonyms)
            if sense > 1:
                Disambiguate the meaning of the word by WordSense Disambiguation using K-mean
        clustering
                for synonym in same_cluster_with_target_word:
                        for synonym, frequency in (synonym, frequency_in_BNC(synonym)):
                                most_frequent_word = synonym that has maximum frequency
                                translation = most_frequency_word
    if sense == 1:
        most_frequent_word = synonym that has maximum frequency using BNC list
        translation = most_frequency_word
        if not translation:
                translation = definition in WordNet

    return translation
```

APPENDIX B

PSEUDO CODE OF THE ALGORITHM IN CHAPTER IV

```
Input: Engineering function E
Output: Translated biotext


keyword == raw_input(E)
Find Biological functional term B from keyword
for sentence in Biocorpus:
        if B in sentence:
                save sentence
read sentence
for word in sentence:

        if word not in BNC_8000word_list & stopword_list:
                translation = Translation(word)
        word = translation



def Translation(word):
        if word in WordNet:
                find synonyms
                sense = count(synonyms)
                if sense > 1:
                                Disambiguate the meaning of the word by WordSenseDisambiguation using
                                  K-mean clustering
                        for synonym in same_cluster_with_target_word:
                                        for synonym, frequency in (synonym,
                                                frequency_in_BNC(synonym)):
                                            most_frequent_word = synonym that has maximum frequency
                        translation = most_frequency_word
                if sense == 1:
                                most_frequent_word = synonym that has maximum frequency using BNC
                                                                list
                        translation = most_frequency_word

        elseif word in ITIS:
                translation = commonly used name or species name in ITIS database

        else:
                Find definition sentence in Wikipedia article
                Extract Noun Phrase(NP) from the definition sentence
                translation = NP
                if translation:
                        break
                else:
                        Find definition from the WorNik
                        if definition in WordNik:
                                translation = definition in WordNik
                        else:
                                translation = word


        return translation
```

APPENDIX C

PREFIX FREQUENCY OF COMPOUND WORDS IN BIOCORPUS

prefix	count	prefix	count	prefix	count
non	28	mid	6	out	1
re	26	immuno	6	mini	1
co	24	down	6	milli	1
un	18	thermo	4	mega	1
sub	17	hydro	3	iso	1
de	16	extra	3	intro	1
pre	14	chemo	3	infra	1
inter	14	up	2	end	1
hyper	14	radio	2	cross	1
over	10	pro	2	cover	1
poly	8	photo	2	con	1
micro	8	intra	2		
endo	8	homo	2		
auto	8	fore	2		
anti	8	ex	2		
multi	7	di	2		
mono	7	bi	2		
mis	7	-like	2		
under	6	uni	1		
post	6	semi	1		
				total	320

APPENDIX D

PSEUDO CODE OF THE ALGORITHM IN CHAPTER V

```
Input: Engineering function E
Output: clustered documents and morphologies

w = constant in penalty function
keyword == raw_input(E)
Find Biological functional term B from keyword
for paragraphs in Biocorpus:
        if B in paragraphs & POS(B) == verb:
                save paragraphs
read paragraphs

for one_paragraph in paragraphs:
for word in paragraph:

        if POS(word) == nouns:
                if noun.category(word) == noun.shape, noun.attribute or noun.artifact in WordNet:
                        save word as morphology
for i in range(len(paragraphs)):
        for j in range(len(morphology)):
                co-occurrence matrix[i][j] == frequency of j-th morphology in i-th paragraph

    ## Apply penalty function to co-occurrence matrix
    distance = sentence distance of morphology and functional verb
    co-occurrence matrix [i][j] == co-occurrence matrix [i][j] * w^(distance)


weighted_matrix =Apply logTF-IDF to co-occurrence matrix
## Apply svd here
U, S, VT = svd(weighted_matrix)
## U ;term matrix, S= eigenvalue matrix V=document matrix

##Find dimensions
for i, eigenvalue in enumerate(S):
        if sum(eigenvalue/sum(S)) > 0.5:
                S[i]=0   #zeroing out eigenvalues
        else:
                save eigenvalue in eignevalues
                pass   #Keep eigenvalues
dimension =len(eigenvalues)
U' =U*S
V' =S*V

# cluster U' using k-means clustering
cluster_num = apply silhouettes

## Apply EM algorithm
morph_clusters =EM(U')
```

```
## group paragraphs
group={}
for k in range(len(morph_clusters):
for one_cluster in morph_clusters:
for paragraph in paragraphs:
if len( morph in paragraph for morph in one_cluster) > p :
        group{k}.append(paragraph)
```