8-2013

Using Ontology-Based Approaches to Representing Speech Transcripts for Automated Speech Scoring

Miao Chen

ABSTRACT

Text representation is a process of transforming text into some formats that computer systems can use for subsequent information-related tasks such as text classification. Representing text faces two main challenges: meaningfulness of representation and unknown terms. Research has shown evidence that these challenges can be resolved by using the rich semantics in ontologies. This study aims to address these challenges by using ontology-based representation and unknown term reasoning approaches in the context of content scoring of speech, which is a less explored area compared to some common ones such as categorizing text corpus (e.g. 20 newsgroups and Reuters).

From the perspective of language assessment, the increasing amount of language learners taking second language tests makes automatic scoring an attractive alternative to human scoring for delivering rapid and objective scores of written and spoken test responses. This study focuses on the speaking section of second language tests and investigates ontology-based approaches to speech scoring. Most previous automated speech scoring systems for spontaneous responses of test takers assess speech by primarily using acoustic features such as fluency and pronunciation, while text features are less involved and exploited. As content is an integral part of speech, the study is motivated by the lack of rich text features in speech scoring and is designed to examine the effects of different text features on scoring performance.

A central question to the study is how speech transcript content can be represented in an appropriate means for speech scoring. Previously used approaches from essay and speech scoring systems include bag-of-words and latent semantic

analysis representations, which are adopted as baselines in this study; the experimental approaches are ontology-based, which can help improving meaningfulness of representation units and estimating importance of unknown terms. Two general domain ontologies, WordNet and Wikipedia, are used respectively for ontology-based representations. In addition to comparison between representation approaches, the author analyzes which parameter option leads to the best performance within a particular representation.

The experimental results show that on average, ontology-based representations slightly enhances speech scoring performance on all measurements when combined with the bag-of-words representation; reasoning of unknown terms can increase performance on one measurement (cos.w4) but decrease others. Due to the small data size, the significance test (t-test) shows that the enhancement of ontology-based representations is inconclusive.

The contributions of the study include: 1) it examines the effects of different representation approaches on speech scoring tasks; 2) it enhances the understanding of the mechanisms of representation approaches and their parameter options via in-depth analysis; 3) the representation methodology and framework can be applied to other tasks such as automatic essay scoring.

USING ONTOLOGY-BASED APPROACHES TO REPRESENTING SPEECH TRANSCRIPTS FOR AUTOMATED SPEECH SCORING

by

Miao Chen

B.S., Peking University, 2005

Dissertation

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in *Information Science and Technology*.

Syracuse University
August 2013

ACKNOWLEDGEMENTS

I owe many thanks to many people that helped me enormously through my dissertation: my advisor Prof. Jian Qin, who led me throughout the once tough time of my doctoral life, taught me to be a researcher, and put through a great deal of time on my thesis; my committee members professors Bei Yu and Howard Turtle, who provided me invaluable input not only to my dissertation but also my research in general; Dr. Klaus Zechner, to whom my special thanks go to, my mentor at Educational Testing Service when interned there, kindly provided data set from ETS, and discussed every detail of my thesis.

I started my PhD work from 2005 with the goal of doing some research in information retrieval, while ending up in conducting research in natural language processing and ontology. I am pleased that the current work is not very far from my goal 8 years ago, and I really enjoy research in text analytics.

During my thesis writing time, I received help from many people, in life and in research. Prof.Beth Plale, whom I worked for at Indiana University, generously allowed me enough time to finish up the thesis. Prof. Nancy McCracken has helped me in research for quite several years and I really appreciate her suggestions on my research.

My PhD journey has been a long one, but it is a good journey and I learned many things. I especially thank my husband Xiaozhong Liu, without whose strong support and love this dissertation would have been impossible. I am grateful to my parents, who are always in upright spirit, and my parents in law, who helped take care of my daughter during the busiest dissertation writing time. Lastly, this is also dedicated to my daughter Tiana Liu, whose smile is the best gift when waking up in the morning.

CONTENT TABLE

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1. INTRODUCTION

1.1 Representation in Information Science: A General Perspective

This study devotes effort to the application of ontologies for text representation. Traditional text representation approaches present two major challenges: meaningfulness of representation and unknown terms. Meaningfulness of representation denotes the conveyance of representation units, such as words and phrases, and the choice of appropriate representation units to express maximum semantics. Unknown terms refer to situations in which terms from external documents do not occur in the existing corpus, which makes it difficult to decide their importance to the existing corpus. The study proposes to employ ontologies to resolve these challenges through ontology-based representations. In order to quantitatively evaluate the performance of ontology-based text representation approaches, it is necessary to identify a use context including text representation outputs and then evaluate how the ontology-based representation would affect system performance in the use context. The author chooses content scoring of second language speech as the context, because meaningfulness of representation and unknown terms are also existing challenges to content scoring. In terms of evaluation, the performance metrics of speech scoring systems are used, and moreover, performance of on traditional representations and ontology-based representations are compared.

In second language speaking tests, test outputs are assigned grades to reflect the language learners' language ability, and automatic systems have been developed to facilitate the grading of speech. In the general sense, this study aims to enhance text representation with ontologies; in the chosen context, it tackles a second language

assessment problem through ontology-based representations adopted from methodologies and approaches in information science.

Automatic content scoring of speech primarily uses natural language processing approaches. The main challenge in scoring the content of spontaneous speech lies in its unpredictability compared to the speech generated from speaking tests such as read-aloud items. For highly predictable speech (e.g. read-aloud items), the content can be accurately recognized, even for non-native speech, and thus the content scoring tasks becomes a string matching problem with no need of considering its meaning. On the contrary, since spontaneous speech is unpredictable, it becomes critical to have meaningful representation of the content to determine the scores.

This study takes a new approach toward the problem of content scoring of spontaneous speech by viewing it from the perspective of information representation, an important aspect of information science. By applying this new framework, the problem is restated in a different way and, accordingly, new approaches to automatic speech scoring can be proposed. It represents speech transcripts and content by using novel ontology-based representations.

This chapter: first, provides an overview of information representation, a major conceptual information science approach; second, reviews text representation for information resource content and discusses its challenges; third, proposes ontology-based representations to resolve representation challenges; fourth, discusses, in greater detail, ontology-based representations in the particular context of automatic speech scoring; and fifth, presents a research framework, research questions, and contributions and implications.

Information representation has been a core issue in information science since its founding. Before the digital era, when information science was within the scope of the library environment, representation issues were addressed in various ways. There are two aspects of library use: human and information resources. An interface is needed to connect them and representation serves as this interface. Representation helps describe resources in meaningful ways, which facilitates information uses such as: retrieval, visualization, browsing, sharing, and discovery. Because many library resources are evolving into a digital format and the scope of information resources is expanding, representation becomes more and more important as this interface.

Information representation has been involved in the broad span of the information science field. In library catalogs, books are described by metadata, such as “author” and “title” information; in an information retrieval system, a document is usually represented by its words and frequencies; in an ontology, knowledge is represented by concepts and semantic relations; in social media websites, users and their relations can be represented by vertices and edges. From the viewpoint of information retrieval, information representation refers to “the essence or the subject content of the document via a certain approach although the end product can take a variety of forms” (Chu, 2003, p.25). This definition points out the essential task of information representation and indicates that such representation can result in different styles. Buckland (1991) claims that representable information is divisible into four categories: data, text and documents, objects, and events. Usually, representation shifts from one category to text or data, however exceptions to this exist as well.

Given the breadth of information science and corresponding range of information tasks, the coverage of information representation expands beyond Chu's definition. The *things* to be represented still fall into Buckland's (1991) five categories, namely, data, text, documents, objects, and events, but the range expands. In addition to documents and library objects and others in the traditional library context, there are diverse information resources, such as images, videos, blogs, social tags, tweets, social messages, scientific data sets, and speech transcripts. Speech transcripts, the last resource listed, are the focus of this study.

Representing information has become a precondition for many information-related tasks, including: information retrieval, information visualization, web browsing, and information sharing. Thinking of tasks from a representation perspective can be called "representationist", with the underlying assumption that representation is a prerequisite for the information-related task and has significant effects on that task. Under this large theory framework, this study is an attempt to tackle the task of content scoring of speech, within which further facilitates the task of automated speech scoring.

Three questions are essential to information representation and, given an information-related task, can be answered according to task content and used to facilitate the task. The first,and most important question is: what is the purpose or context of the representation? The answer to this question explores the context of representation. The same information resource may be represented in various ways, while the context of its use will determine the appropriate approach from many possibilities. For example, suppose the task is to represent an academic paper in digital format. If the purpose is to label document information, such as creator and title, then a

metadata approach, such as the Dublin Core schema, is appropriate. If the purpose is to describe content, then document representation approaches, such as the bag of words approach, can be used.

The second question is: What information resource is to be represented? It asks for a resource unit to be represented in a way suitable for the context. For example, an information resource can be a physical book, a webpage, or an online message or part thereof. The representation can serve as container that delineates extrinsic information such as the creator of the resource, size of the resource, time it was created; and the representation can also describe topics, namely, what the content is about. The transformation from resource units to their representations can better facilitate people's understanding of and interaction with the resources. For representation of the meaning of resource content, it is important to clearly identify the resource unit to support the representation task.

Finally, how should the information resource be represented? This question concerns the approach used to describe information resources and, as mentioned above, there are various options to choose from. Answers to the first define the purposes of representation, and those to the second question identify the kinds of description of resources suitable for the purposes. Answering both first and second questions can help narrow down the appropriate approaches to be used.

This study deals with how the third question can be answered given responses to the first two questions. The status of the three questions constructs the research setting. In the following sections, the motivation and purpose of this study will be addressed in more detail, however the research setting is briefly summarized here. That is: the

purpose or context is content scoring of speech, the information resources are speech transcripts, and the research questions surround how speech transcripts can be represented for this particular purpose.

Representation approaches for information resources can be roughly grouped into two categories: description-based and content-based. Description-based approaches aim to describe information resources by their exterior characteristics, such as: date, author, or owner of an information resource. In contrast, content-based approaches focus on the interior part of the resource, such as: strings in text documents, sound in spoken documents, and pixels in image documents.

Representing resources involves the use of language with manual, automatic, or semi-automatic processes. Using indexing as an example approach for representing document content, people can manually label the topics of an information resource or computers can automatically analyze content features using natural language processing techniques. The language aspect addresses language or vocabulary used for representation, some being controlled and some others being free text. Continuing with the indexing example, representation can either employ controlled language, such as vocabulary, categories, and ontologies to represent resource content, or free language, such as words from the content when actually representing a resource.

This study will employ content-based representation approaches with speech transcripts being the resource and content scoring of speech the context. The speech transcripts will be represented via both accepted and proposed approaches and the effects of different representations will be evaluated afterwards.

1.2 Text Representation: Overview and Challenges

Information representation can refer to every possible aspect of an information resource, including content representation. This study will focus on content representation of information resources, which is often referred to as “text representation”. The areas of text processing, such as natural language processing (NLP) and information retrieval (IR), frequently deal with content representation of text documents, therefore the representation approaches for these areas can have important implications on content representation problems. The data used for this study consists of speech transcripts, which are a special type of text document and can be considered as a special case of text representation.

Text representation has been an important topic in research fields such as information retrieval, natural language processing, and text mining. A variety of text representation methods have been proposed in previous literature, including Salton, Wong, and Yang (1975), Deerwester, Dumais, Furnas, Landauer, and Harshman (1990), Lewis (1992), Kaski (1997), He, Cai, Liu, and Ma (2004), Arguello, Elsas, Callan, and Carbonell (2008), Hotho, Staab, and Stumme (2003a) and many more, as discussed in the literature review chapter. These methods are rooted in different views of text documents and are used to address different situations. These methods comprise a diverse source of approaches and implications, which are useful for developing a new type of representation through this study.

Many of the existing text representation approaches from NLP and IR are statistically and corpus based. One prevalent representation approach is the bag-of-words approach, in which a document is represented by word vectors. Document vectors and representation units by words are constructed through the extraction of

statistical information like term frequency or weight. Other approaches use latent variables as representation units, which are statistically mined from a corpus to represent documents, for example the Latent Semantic Analysis (LSA) and Latent Dirichlet Allocation (LDA) algorithms. Their general assumption is that there is hidden structure in documents and statistical analysis assists in revealing the structure, either as latent concepts or topics. Empirical experiments have shown that these statistical approaches perform well (Deerwester et al., 1990; Blei, 2011). However, current computer systems, such as search engines, employing these approaches are still far from fully understanding natural language text; new approaches are needed to express richer semantics of documents. This author's purpose is not to turn away from statistical approaches, but rather to focus more on semantics and concept-level representation to overcome the problems caused by statistical approaches.

Approaches to statistical representation exhibit two major challenges. The first is the meaningfulness of the representation. In bag-of-words representation, words with similar meanings are treated as different dimensions and are thus independent of one another; the relationships between similar words are not reflected in this type of representation. Similar words should be grouped together as one dimension to construct a meaningful semantic space. As a result, similar words would be treated equivalently to ensure consistent credit in content scoring. It would thus seem reasonable to represent documents by groups of words or concepts as representation units. In fact, concepts are similar to groups of words, since a concept subsumes several synonyms. Statistical approaches generate latent concepts or topics; however, it is not easy to interpret latent variables, since they make sense as statistics rather than

semantics. In other words, meanings are not explicitly revealed from statistical results. In contrast, ontologies contain explicit domain knowledge, including concepts and semantics, and can construct a semantically meaningful space for document representation.

The second challenge to statistical approaches is unknown terms. The reliance of NLP on corpus (Linckels & Meinel, 2011) raises issues of unknown terms, because statistical representation approaches generally use a corpus as the basis for building the representation space. For example, vector dimensions of the bag-of-words approach come from words within the corpus and latent concepts from semantic analysis come from decomposition of a corpus-based matrix. These approaches explore the corpus to extract local knowledge. This research seeks to illustrate that global knowledge, like ontologies, can complement weaknesses of local knowledge extracted from a corpus. A corpus is unlikely to contain every possible term in the domain, and perhaps an alternative representation method can be used to mitigate the problem. Ontologies as knowledge bases can provide knowledge about unknown terms to complement the insufficient coverage of the corpus and are thus worth exploration and testing.

The two aforementioned challenges make statistical approaches limited for text representation in general and in speech transcript representation in particular. The next section will articulate on ontology-based representation and why it might work for speech transcripts representation.

1.3 Ontology-based Representation: A Complement to the Challenges

Motivated to address these challenges to text representation, this study proposes to use ontology-based representation. A well-accepted definition of ontology in the artificial intelligence field is “ontology is a specification of a conceptualization” (Gruber, 1993). Ontology is used for knowledge representation, and knowledge of a domain is conceptualized by concepts and relations (conceptualized), and is then expressed in formal language (specification). The author claims that ontologies can help tackle the problems due to their own characteristics. More specifically, the author considers ontologies containing fairly large number of instances of concepts (or domain vocabulary) are feasible for text representation because we need to match concepts in text to ontology concepts. Examples are WordNet, Wikipedia, and UMLS ontologies. Abstract ontologies, the ones containing abstract and high-level classes without domain vocabulary, such as the SWEET ontology and SKOS ontology[1], are therefore not suitable for text representation task.

Firstly, ontologies contain concepts and semantic relationships defined by people, making the elements meaningful and accurate. Using ontological concepts for representation can avoid problems of synonymy and homonymy, which are prevalent in bag-of-words representation (Bloehdorn et al., 2011). Ontologies can be used to group synonyms and related words in the same dimension to form more meaningful document representations. This will result in vectors of ontology concepts, which are a concept-level representations. Ontology-facilitated representation has been employed in tasks such as clustering, classification, and information retrieval (Bloehdorn & Hotho, 2004;

[1] SWEET ontology http://sweet.jpl.nasa.gov/2.2/
SKOS ontology http://www.w3.org/TR/skos-primer/

Hotho et al., 2003a; Hotho, Staab, & Stumme, 2003b; Muller, Kenny, and Sternberg, 2004; Wang, McKay, Abbass, & Barlow, 2003; Zhang, 2009).

Secondly, semantic relations defined by ontologies connect relevant concepts and organize them into trees (i.e. WordNet) or graph structures (i.e. Wikipedia). Since paths usually exist between two individual concepts, ontologies can support inferences about related concepts by using the paths and concept nodes between them. The inference potential of ontologies can help resolve the “unknown terms” challenge of statistically based representations. One possible resolution is to infer the importance of unknown terms based on concept similarity or relationship between unknown terms and known terms. As the importance of known terms is known, the importance of unknown terms can be inferred by integrating the importance of known terms and the similarity between the known and unknown terms. Methods of computing concept similarity in ontologies have been proposed by Lin (1998), Pedersen, Patwardhan, and Michelizzi, (2004), Resnik (1999), and Strube & Ponzetto (2006).

The two previous features of ontologies complement the statistical approaches. Bloehdorn et al. (2011) discuss the legitimacy of combining statistical approaches (data driven and inductive) and ontologies (semantic and knowledge based) to facilitate text mining. They claim the two are good complements to each other because the former offers learned patterns from real world data and the latter provides structured and encoded world knowledge. In this study, the author identifies a context in which ontology-based representation can be applied and examines the influence of ontologies in context.

1.4 The Test Bed: Content Scoring of Speech

As mentioned earlier, in order to examine the effects of using ontologies in information representation, a context needs to be defined for empirical evaluation. Content scoring of speech is selected as the context to be examined as the test bed for the study. In this particular context, speech transcripts are the information resources and speech content is the information to be represented. As a type of text document, speech transcripts also experience the two challenges of text representation: meaningfulness of representation and unknown terms. Similarly, ontology-based representations can be a proposed solution to the challenges of speech transcript representation. Therefore, empirical study on representation of speech transcripts has implications for how ontologies can affect text representation in general.

Content scoring of speech, the examination context, is briefly discussed and then linked to ontology-based representation. From the aspect of language assessment, this context belongs to the category of automatically scoring speech generated by second-language speakers. With more and more people taking second language tests, such as TOEFL® (Test of English as a Foreign Language) and IELTS™ (International English Language Testing System), adopting automated scoring techniques has become an attractive idea for purposes of efficiency, productivity, and objectivity.

Speaking is an important aspect for assessing second language speakers' proficiency, along with listening, reading, and writing (Bachman & Palmer, 1996). When giving a speaking test in a computer-mediated environment, test-takers' responses are typically stored as speech files. These files can be considered to contain two layers: sound and text. The sound contains the acoustic features of speech, which are used to assess speaking proficiency in existing automated speech-scoring systems (Dodigovic,

2009; Zechner, Higgins, Xi, & Williamson, 2009). However, the text features, which embody speech content, are not well addressed or utilized in scoring systems because of low accuracy of automatically transcribing spontaneous non-native speech to text. That is to say, speech scoring from the content side is less explored than the acoustic side. However, the representationist view has not been applied to speech scoring before and therefore forms a good test bed for the study.

In order to perform content scoring on speech, content has to be converted to a representation format that enables automated scoring. In other words, some processing needs to be performed on documents (i.e. speech transcripts) and the representation of documents serves as an interface between human and content. On the one hand, speech scoring researchers' views of documents influence their choice of representation approach. For example, if researchers consider documents as strings of words, then they will likely adopt bag-of-words representations. On the other hand, document contents are represented in a computable way, so that further computation (e.g. content scoring) can be performed. Therefore, the representation integrates researchers' views as well as content from documents. Furthermore, its functionality, as the interface between human and content, will facilitate automatic speech scoring.

As previously stated, ontologies can resolve both of the challenges of speech transcript representation. Moreover, the use of ontologies in this particular context is legitimate. First, building scoring models typically adopt the training-test paradigm, which means that a portion of speech transcripts are used to build a content scoring model and the rest are used for evaluating model performance. Terms, concepts, or topics, along with their importance, are extracted from training transcripts for model

building. There are scenarios when a term occurs in testing transcripts, but does not occur in the collection of training transcripts, which raises the question of how this term should be assessed. Because there is no knowledge about a term from the training transcripts, it needs to be discarded when scoring the test transcripts. This causes information loss and possibly negative effects on scoring results, whereas the rich domain coverage of ontology may complement the unknown term issue.

Second, using ontologies for automated speech scoring may help deal with some issues in this research area as well. Existing speech scoring systems predominantly consider acoustic features, such as fluency, pronunciation, and prosody (Chen & Zechner, 2011), while content features are frequently overlooked. Hence, adding content features will expand the coverage of scoring models and may improve performance. Ontology-based representation uncovers concepts embedded in speech transcripts, which would have not been detected by other approaches. It thus seems to be a more appropriate approach. In automated speech scoring, factors like content relevance and topic relevance are important for measuring speaking proficiency. Content features are indicators of content and topic relevance and can be processed by computers. Words can reflect content to an extent; however, the concept level is closer than the word level to the topic level. Ontology-based representation may provide a better foundation for measuring topic and content relevance in the semantic and conceptual space. Therefore, ontologies meet the need to measure content and topic relevance for speech scoring, as well.

The use of ontologies is therefore feasible from both the perspective of text representation and speech scoring. Evidence collected for speech transcript

representation has implications for text representation in general and the framework of generalization and specification is discussed in the next section.

1.5 Research Framework and Design

1.5.1 Conceptual Framework

Based on the previous discussion of document representation, ontology, and content scoring of speech, this study will be guided by a conceptual framework as shown in Figure 1. It illustrates how the central research problem guiding this study is generalized and specified. The generalized argument is that ontology may facilitate text representation and further information tasks, which all take place on the abstract level. On the concrete level, the context and test bed of the study is identified as content scoring of speech. The problem becomes how ontologies can facilitate speech transcript representation and specifically support speech content scoring. Experiments will be conducted on the concrete level and results will be analyzed and generalized to answer the research problem at the abstract level.

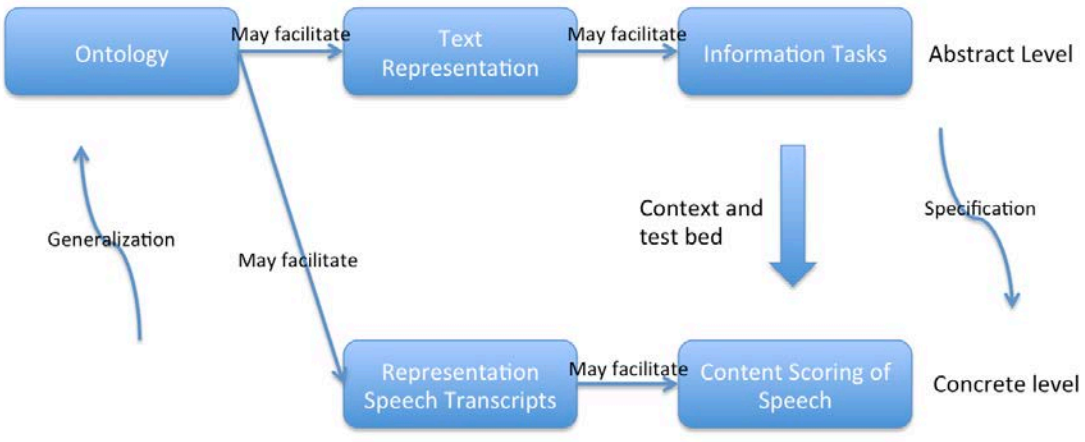


Figure 1. Generalization and specification of the study.

This study adopts the representationist view from information science, as diagramed below (Figure 2) for the concrete level of the study. Test takers generate spoken documents, which are another type of information resource but not the main focus of this study. Then, spoken documents are transcribed to speech transcripts, which are the core information resources of the study. In this specific context, the core question can be phrased as: “How can speech transcripts be represented for the content scoring of speech”?

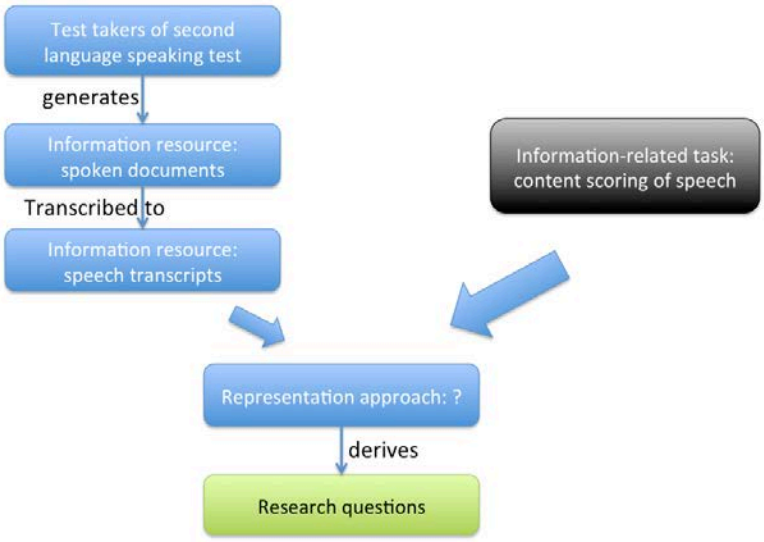


Figure 2. A representationist view of the study.

1.5.2 Research Design

The study follows empirical research paradigms by conducting experiments. Ontology-based approaches will be employed to represent speech transcripts for scoring tasks. They address challenges of statistical representation approach, as well as automated speech scoring issues, as previously stated. While the rationale for the new approach makes theoretical sense, empirical experiments are necessary to examine performance in real information tasks. Performance will be investigated via

observing scoring task outcomes when applied to the content of speech. Two ontology-based representations are proposed and delineated in Chapter 3.

As comparisons, two prevalent representation approaches in text processing are implemented as baselines: bag-of-words and latent semantic analysis. These approaches are frequently used in content scoring of essays, such as in the e-rater® and Intelligent Essay Assessor™ systems (Burstein, 2003; Landauer, Laham, & Foltz, 2003). All the representations are then used to score speech and representation performance is used to evaluate and interpret their effects. The figure below illustrates the research design.

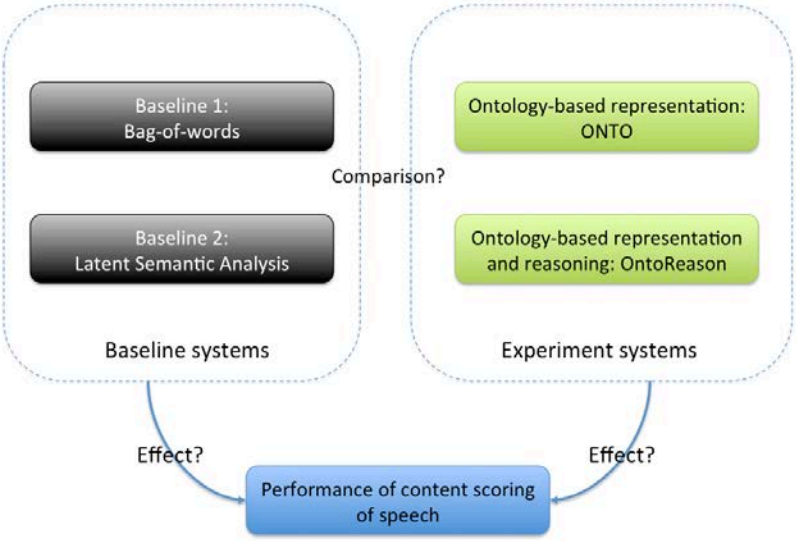


Figure 3. An overview of the research design.

In this experiment design, the effects of different speech content scoring approaches are assessed through three modules: representation, scoring model, and evaluation (in Figure 4). The speech transcripts are: first represented as vectors, then vector outputs are taken as input of the scoring model (machine-learning based), and, lastly, the scoring model performance is measured by the evaluation module as an indicator of the effectiveness of representation approaches.

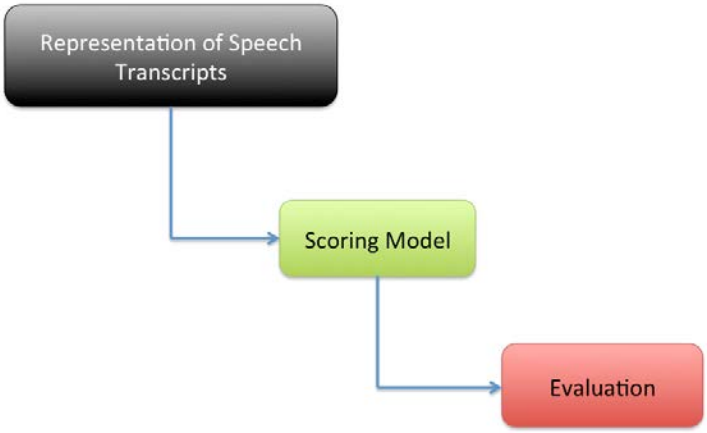


Figure 4. Three modules of the experiment design.

Representation Module

This module is about converting speech transcripts to vector representations by using different representation approaches. The choice of representation approaches can affect the construction of scoring model and further affect evaluation result.

Scoring Model

The scoring model uses machine learning approaches to assign scores to speech transcripts. Though representation is the core issue of the study, scoring model is a critical component because it outputs scores for evaluation purpose. It takes representation results, namely vectors, as input, and learns a classifier from training data. The classifier can be used to assign scores given a test speech transcript.

Evaluation Module

The classifier generated from the scoring model is then used to assign scores to test transcripts. The evaluation module assesses performance of scoring module by comparing machine assigned scores and human assigned scores. Moreover, the evaluation is not only an indicator of performance of scoring model, but it also reflects performance of representation approaches in an indirect way.

1.6 Research Questions

The research questions are formulated based on the conceptual framework and will be answered by experiment results. As reflected in the conceptual framework, two parts are critical to the research: ontology-based representation and content scoring of speech. Therefore the major research question is:

How do ontology-based representations of speech transcripts affect the performance of content scoring of speech in automated speech scoring systems?

This is an overview question that addresses what effect ontology-based representation may have on automated speech scoring systems and guides specific follow-up questions. The ontology-based text representations are evaluated for their influence on speech scoring. Comparisons will be made between ontology-based representations and baseline representations. Prior to evaluation and comparison, it is necessary to determine whether ontology-based representation makes a difference, thus the first specific question posed is:

RQ 1: Does ontology-based representation of speech transcripts perform differently from the baseline approaches in content scoring of speech?

The performance of a speech scoring system will be measured by its predicting ability of speaking proficiency (e.g. holistic scores by human raters). The transcripts will be represented by various approaches and these representations will be used to build content scoring models to predict speech scores. Evaluation metrics, such as F-measure and correlation, can then be used to analyze how well the scoring system can predict speaking proficiency. In this way, system performance will be measured and used to compare differences between effects of baseline and ontology-based representation approaches.

RQ2: How does representing speech transcripts by ontology concepts affect the performance of the content scoring of speech?

To answer this question, ontology concepts will be used as vector dimensions to turn a document into a vector for representation. This does not consider inference of the importance of unknown concepts and addresses only the effects of concept-level representation. The baseline systems, bag-of-words and latent semantic analysis approaches, will be compared against concept-level representation. Similar to RQ 1, scoring models computed from the representation will evaluate performance based upon their predicting ability. The next question takes into account the effect of unknown concepts:

RQ 3: To what extent does inferring the importance of unknown concepts affect the performance of the content scoring of speech?

Besides using ontology concepts for representation, concept connections within the ontology can be used to infer the importance of unknown concepts in testing transcripts. As a result, including unknown concepts as additional dimensions enriches representation of testing and training transcripts. Thus, this constitutes another type of ontology-based representation. This new representation is compared against the purely ontology-based one (without inferring), in order to assess the effect it has on content scoring of speech. The performance measurement is similar to that of RQ 2.

These research questions guide the four hypotheses proposed in section 3.3. In other words, the hypotheses are the bridge between research questions and experimental design, so that the experimental results can answer the research questions via the linkage.

1.7 Contributions and Implications

This study tackles the challenges of text representation through ontology-based representation, using content scoring of speech as a test bed. It proposes ontology-based representation for speech transcripts and evaluates its effect in the context of automated speech scoring. It provides new applications for ontologies in text processing, while focusing on one particular task. Content, a less discussed aspect of automated speech scoring, is addressed in this study, to close the literature gap. Meanwhile, it will also enhance understanding of concept-level representation and concept semantic space.

Besides answering the research questions, this study also has implications for the use of ontology-based representations in content scoring of essays. Test takers' essays, which are similar to speech, are another output of second language testing. Speech is most similar to essays when disfluencies (e.g. repetitions, false starts, pauses) have been removed. That is to say, the effect of ontology-based representation on essays can be estimated to some extent by considering the effect on speech transcripts. Finally, as speech transcript is also a type of text document, this study has implications for text document representation using ontologies as methodology.

CHAPTER 2 LITERATURE REVIEW

In this chapter, I review three research areas critical to the topic of the study: document representation, second-language assessment, and the use of ontologies in text processing. Studying existing approaches of document representation, which are mostly statistical, not only outlines the area but also provides a foundation for the baseline system in the methodology chapter. The author proposes using ontology for document representation, as it complements statistically-based representation. Thus it is important to review definitions and characteristics of ontology, as well as methods of use in text processing. As the information-related task in this study is automated speech scoring, aspects relevant to this task are studied, such as theoretical grounds for second-language assessment and practices in automated scoring.

General document representation approaches are surveyed in section 2.1, theoretical aspects and automated scoring methods of second language assessment are presented in section 2.2, and ontology and its use in text processing are discussed in section 2.3.

2.1 Document Representation in General

This section reviews document representation approaches, of which two important methods are stressed: bag-of-words and latent semantic analysis approaches. These approaches are used as baseline systems in the experimental design and therefore their details are addressed here. Other approaches, along with document representation in essay-scoring systems, are briefly discussed.

2.1.1 Bag of Words

The most common text document representation is bag-of-words representation. As a widely used approach for information retrieval, text mining, and natural language processing, this representation method treats documents as a set of words, while more complex information, such as phrases and semantic concepts, and structural information, such as word order and sentence order, are not considered.

The bag-of-words representation is closely related to the vector space model (VSM) in information retrieval, proposed by Salton et al. (1975). The basic idea of VSM is that a document is a vector of words, with each dimension of the vector standing for a single word. It assumes that words are independent from each other. The vector space is constructed by all words in the document collection except stopwords, which are function words with little meaning (Croft, Metzler, & Strohman, 2010, p.90). A given document can be represented by a vector via constructing a vector space and positioning it in the space. For example, suppose a document contains the words “Green tea ice cream is ice cream;” its vector representation is (green, tea, ice, cream, is), by writing out all the unique words. Furthermore, once document vectors are obtained, they can be used to measure distance between documents by utilizing similarity measurements, such as cosine similarity, as the basis for further text analysis.

When converting a document into a vector of words, one needs also to assign values to the word dimensions. Each word, as a dimension, possesses a value in the vector and together all of the words comprise a vector. The value of a dimension indicates the importance of the target word to a specific document, which is also referred to as the “weight of the dimension”. Methods to determine the heuristic or

empirical weights of word dimensions have been investigated for years in research fields like information retrieval and text mining.

The weight of a word can be expressed at two levels: the document level and the corpus level. The former weight is the importance of this word in the document; the latter weight is its overall importance in the document collection. Document-level weight is usually measured by Boolean or Term Frequency (tf) related information. “Boolean” indicates the presence of a word in the local document, while TF describes its frequency or normalized frequency in the document.

For the corpus-level weight, the idf (inverse document frequency) measure is often used. This assumes that a word is more important if it occurs in only a few documents in the collection and, conversely, if it appears in numerous documents, that it is relatively unimportant. A word’s IDF value can be calculated by dividing the total number of documents by the number of documents in which it occurs.

Given a document collection, we can compute a word’s weight on the document and corpus levels in terms of multiplying tf and idf, which balances the word’s importance locally and globally. Most of the weighting schemas are variations of the tfidf schema (Croft et al., 2010, p.241).

Normalization is a factor that is often considered when assigning weight to words. Raw weights raise problems when comparing different documents; normalization is usually used to adjust word weights to the same scale. Taking document length as an example, because document length is variable, a word with a high frequency in a document does not necessarily mean high importance since the high frequency can be due to the document’s long length; similarly, low frequency does not definitely mean a

word is insignificant. Therefore, normalization can be used to reduce the effect of document length on term weighting, for example, by dividing term frequency by document length. Other normalization methods include dividing term weights by the maximum term frequency of the document and by the Euclidean length of the document vector, which refers to the square root of summed squares of each term weight.

Several weaknesses of bag-of-words representation have been identified. First, it assumes words are independent from each other, which is not true in the real world. For example, the word “cake” tends to co-occur more frequently with the word “bake” than with other irrelevant words like “bicycle.” Second, multi-word expressions are missing from the representation since they are broken into distinct words. For example, the phrase “part of speech” means the syntactic roles of words in a sentence, but the meaning is lost in breaking the multi-word expression to three single words. Third, the bag-of-words approach causes synonym and polysemy problems, meaning synonymous words are not mapped to the same dimension in the vector and a word with multiple meanings is mapped to the same dimension. Fourth, the representation lacks generalization of words. For example, words that are similar to each other should be grouped under their hypernyms to form the same dimension (Bloehdorn & Hotho, 2004; Croft et al., 2010, p.452).

2.1.2 Latent Semantic Analysis

Latent Semantic Analysis (LSA) initially appeared in information retrieval literature, manifested in papers such as Deerwester et al. (1990). The term is often called Latent Semantic Indexing (LSI) in the information retrieval field, where LSA is

applied as a document indexing approach. LSI and LSA are often used interchangeably to refer to their use in document representation.

LSA was developed to resolve some issues emerging from word-based text processing, like the bag-of-words approach. Classical retrieval systems employ words as indexing units, which has two major limitations: synonym and polysemy problems (Deerwester et al., 1990). Synonym problem means that words with similar meanings are indexed separately and therefore documents containing synonyms of query terms are not matched. For example, if a query includes “lawyer,” documents containing “lawyer” or “attorney” should ideally both be returned since “attorney” shares a similar meaning to “lawyer”. The polysemy problem refers to the phenomenon that a word can have multiple meanings and its particular meaning is determined by its context. Thus, indexing the word as a unit obscures its contextual meaning. For instance, the word “leopard” can mean a type of big cat or an operating system, but it is not possible to disambiguate the sense of the word by word-based indexing. LSA resolves these two problems by changing the indexing unit from words to latent semantic structure.

In addition, LSA reduces the dimensionality of document representation. The bag-of-words representation uses all the terms in a corpus to represent a document, making the number of vector dimensions equal to vocabulary size and thus resulting in high dimensional space. This representation usually contains “noise” from unimportant words and dilutes the importance of informative words.

Since LSA performs matrix decomposition to reduce the dimensionality of documents, it highlights underlying important semantic structures and removes unimportant ones according to matrix analysis. It is also considered to be a vector

space approach to document modeling (Kontostathis & Pottenger, 2006) in the sense that it models documents based on latent concepts in the vector space style.

LSA starts with a term by document matrix, in which terms are rows and documents are columns. Given term by document matrix A in which an element A_{ij} is about word i and document j, the value of A_{ij} is determined by word weight in the document, for example the frequency of word i in document j. In this way, a corpus can be represented by a term by document matrix. Then Singular Vector Decomposition (SVD), a type of matrix decomposition technique, is performed on the matrix to decompose it into three matrices. SVD is a decomposition approach similar to factor analysis (Deerwester et al., 1990), both aiming to reduce dimensionality of data and detect latent concepts within the data. The product of three resulted matrices from SVD reconstructs the original matrix A:

$$A = TSD^T$$

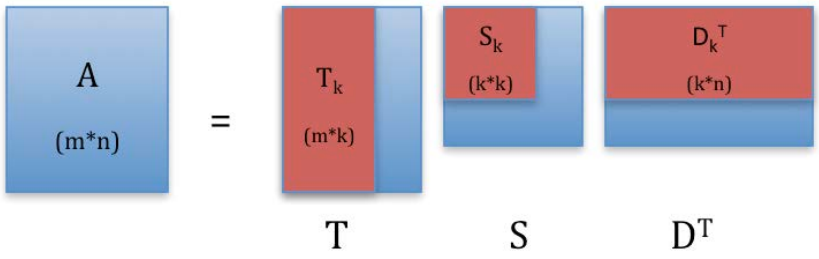
where the T and D matrices are orthogonal matrices and the S matrix in the middle is a diagonal matrix (Deerwester et al., 1990). Matrices T and D are about eigenvectors of terms and documents respectively, and S is the eigenvalue. For example, suppose matrix A is an m*n matrix, representing m terms and n documents, and the three matrices are m*r, r*r, and r*n respectively, where r is the rank of the matrix A. The original matrix and resultant matrices are shown below:

A		T		S		D^T
$\begin{bmatrix} a_{11}\ a_{11} \dots\ a_{1n} \\ a_{21}\ a_{22} \dots\ a_{2n} \\ \dots \\ a_{m1}\ a_{m2} \dots\ a_{mn} \end{bmatrix}$	=	$\begin{bmatrix} t_{11}\ t_{11} \dots\ t_{1r} \\ t_{21}\ t_{22} \dots\ t_{2r} \\ \dots \\ t_{m1}\ t_{m2} \dots\ t_{mr} \end{bmatrix}$	$\begin{bmatrix} s_{11}\ 0 \dots\ 0 \\ 0\ s_{22} \dots\ 0 \\ 0\ \dots\ 0 \\ 0\ 0 \dots\ s_{rr} \end{bmatrix}$		$\begin{bmatrix} t_{11}\ t_{11} \dots\ t_{1n} \\ t_{21}\ t_{22} \dots\ t_{2n} \\ \dots \\ t_{r1}\ t_{r2} \dots\ t_{rn} \end{bmatrix}$	
m*n		m*r	r*r		r*n	

The SVD process is preparation for the dimensionality reduction step. Matrices T, S, and D^T can be reduced and the product of the reduced matrices reconstructs the original matrix A with reasonable accuracy. After ordering the singular values of matrix S, the first k largest values of S are maintained and the others are disposed (Deerwester et al., 1990), and consequently, the number of columns of matrix T and the number of rows of matrix D are reduced to k as well:

$$A = T_k S_k D_k^T$$

where k is an integer that must be determined empirically, based on its influence on task performance. Dimension reduction is as follows:



The resultant factors from SVD can be “thought of as artificial concepts” (Deerwester et al., 1990), for example, the columns of T_k and rows of D_k^T. These factors or dimensions are the latent semantic structure revealed by the SVD statistical approach. They are called “latent concepts” to reflect their nature. Then matrix T_k can be seen as representing terms by the latent concepts (k concepts in total in the above example), and matrix D_k, the transposed matrix of D_k^T, is considered the representation of documents by the latent concepts. The terms and documents are represented in vector space style.

Because terms and documents are represented by vectors of latent concepts, it is feasible to compute similarity between documents, between terms, and between

documents and terms for further textual analysis. Besides documents in the original matrix A, it is also possible to compute the LSA vector for any new document. Given a new document, its bag-of-words vector is d, and its vector of latent concepts can be derived by formula $d^T T_k S_k^{-1}$ (Garcia, 2006).

One challenge of LSA lies in the difficulty that the approach has in interpreting latent concepts or factors derived from SVD. The vectors used to represent documents are eigenvectors generated by statistical process, making it difficult to relate them to human mental concepts. Deerwester et al. (1990) mention that they do not intend to verbally describe the meaning of the factor but that they do represent documents and queries in a more reliable and reduced semantic space. Kontostathis & Pottenger (2006) propose high-order co-occurrences to facilitate interpretation of LSA results. Given a corpus, a matrix of term-to-term co-occurrence can be drawn and from it new term-to-term matrix can be produced from LSA results. They provide mathematical proof that: if there is non-zero co-occurrence between a pair of terms in the co-occurrence matrix computed from LSA matrices, then there exists a connectivity path between the two terms in the original co-occurrence matrix derived from the corpus. If a term co-occurs with another term in a document, this is called first-order occurrence; if it does not co-occur with another term, but they both occur with a third word, this is called second-order occurrence. Term-to-term matrix from LSA results reveals term similarity between term pairs of second-order and higher order co-occurrences. Also, high-order occurrence between terms, such as second-order occurrence, exhibits strong correlation with LSA performance, as measured by the F measure in information retrieval (Kontostathis & Pottenger, 2006).

2.1.3 Other Representation Approaches

In addition to bag-of-words and LSA, other representation approaches have been investigated and are usually compared with the bag-of-words representation. The bag-of-words approach uses words as representation units, which is a convenient but problematic assumption. It breaks constituent-like-phrases and multi-word terms into single words, which often causes some semantics to be lost or distorted. Thus, it seems to make sense to represent documents by phrases or multi-word terms to avoid this kind of problem. For example, Scott & Matwin (1999) try to represent texts using syntactic relationships, such as noun phrases, and semantic relationships, such as WordNet synonyms and hypernyms. Lewis (1990) employs clusters of syntactic phrases as representation units for text categorization tasks. However, no significant improvement was found when using these representations in their studies (Scott & Matwin; Lewis, 1990).

The LSA approach reveals latent semantics by projecting documents into subspace; however, some information is lost during the process. He et al. (2004) claim that LSA detects the global semantic structure but does not reveal documents' local structure. They have thus proposed locality-preserving indexing to represent the local and discriminative structure of documents. This assumes that documents are in a manifold space, unlike LSA, which assumes that documents are in Euclidean space. The locality-preserving projects are used for dimensionality reduction, whereas SVD is the reduction algorithm for LSA (He et al., 2004). Chen, Tokuda, and Nagai (2003) argue that the distance between the original term vectors and the new space is ignored when projecting documents to the new space. They propose the Differential Latent

Semantic Indexing (DLSI) algorithm to respond to the problem by introducing intra- and extra- document vectors to reflect document difference.

Inspired by the logic behind DLSI, Chen, Zeng, and Tokuda (2006) employ a multi-perspective document representation, as compared to single vector representation. Usually, one document is represented by one vector, like a vector of words, while their study proposes representing documents by multiple vectors. A document is segmented into several subfiles, each of which is used to represent one perspective. The subfiles are parts of the original document and share overlapped content. This multi-perspective representation can be applied to vector style representation, such as bag-of-words and LSA approaches, by using more than one word or LSA based vectors.

The Self Organizing Map (SOM), a neural network unsupervised learning algorithm proposed by Kohonen (Lin, Soergel, & Marchionini, 1991), has been used to reduce document dimensions by projecting document content into a two-dimensional space (Lin et al., 1991; Kaski 1997; Kaski, Honkela, Lagus, & Kohonen, 1998). For this approach, documents are represented by low-dimension vectors and are thus easy to visualize. The distance between documents in the SOM space indicates their semantic similarity.

Representing meaning instead of literal terms is definitely the approach to presenting semantics in documents. Recently, Clarke (2012) proposes a context-theoretical framework to represent meaning through vectors. Clarke claims that there has been a gap between vector techniques for representing the meaning of words, such as latent semantic analysis (the word level), and theories of meaning that usually rely on

logic and ontology (the sentence level). He also provides a theoretical framework for meaning representation through vector-based approaches, with the “context as meaning” (meaning comes from context) and develops an algebra to establish a vector-based composition for representing the meaning of strings in text (Clarke, 2012).

2.1.4 Local and Global Representations

Document representation research often addresses issues of range of representation, for example, local and global-based. The issue is reflected in the weighting calculation, which sometimes contains either local or global information. As mentioned in the bag-of-words overview, TF is a local counter, since it is about term occurrence in the local document, and IDF is a global indicator since it considers a word’s importance based on the entire corpus.

For example, blog documents are different from ordinary single documents because a blog is comprised of multiple posts. One issue is whether to represent blogs on the blog or individual post level. Arguello et al. (2008) experimented with two models, large and small, for representing blog posts. The large model represents documents on the blog level by concatenating all posts as one document. The small model takes each post as a document, augmenting the model through normalization by multiplying the probability of a post occurring in the whole blog which provides an advantage over the large model, thereby supporting the large model.

The local and global representations mentioned above are both generated from one or multiple documents. From the perspective of knowledge representation, knowledge is implicitly embedded in text and these representations extract useful information to present text in a more explicit way to computers. On the other hand,

ontologies contain explicit knowledge and are at global level of domain knowledge representation for computational use. General domain ontologies, such as Wikipedia and WordNet, are good sources of conceptual terms and using such terms for text representation may facilitate information related tasks. Ontology-based representation is further discussed in section 2.4.

2.1.5 Dimensionality Reduction

As we can see from bag-of-words representation, the document space that results has high- dimensionality, with the number of dimensions equaling the corpus' vocabulary size. Representation is usually sparse, since word coverage of a document is far less than the vocabulary of the whole corpus. With the inclusion of unimportant words, the high- dimensional space also dilutes the importance of informative words. Therefore, dimensionality reduction can enhance document representations, especially bag-of-words representation, to alleviate the high-dimensional problem.

Dimensionality-reduction methods can be clustered into two groups: term-based and subspace-based. Term-based methods reduce document vector size by eliminating unimportant words based on the word's importance in: a document, a corpus, or a combination of the two. For example, a word with low occurrences in a corpus can be removed from the corpus vector, or a word with a low tfidf weight can be removed from document representation. Yang and Pedersen (1997) evaluated 5 term selection techniques for categorizing the Reuters and OHSUMED corpora and found that information gain and χ^2 test are most effective in experimental settings. Subspace-based methods are similar to the global methods that project the original document space to a subspace with fewer dimensions and posit the documents in the same

subspace for analysis (Chen et al., 2003). Examples include the projection from term-by-document matrix to LSA matrix and the self organizing map.

2.1.6 Document Representation in Essay Scoring

Most automated essay scoring systems represent essays through either bag-of-words or the latent semantic analysis (LSA) styles. Systems employing bag-of-words representation include the e-rater system (Burstein, 2003; Attali & Burstein, 2006) and the experimental system described in Larkey & Croft (2003). Representation in the BETSY system (Bayesian Essay Test Scoring System) also encompasses words, such as frequency of content words; specific phrases are also involved in the essay representation (Dikli, 2006). The exemplary system employing LSA representation is the Intelligent Essay Assessor system, which performs latent semantic analysis on training essays and then projects them into the vector space of latent concepts (Landauer et al., 2003).

E-rater and IEA both use vector style representation, in which estimating weights of vector dimensions plays an important part. In e-rater, given an essay, the weight of a word dimension is determined by multiplying the local and global weights. The local weight is the frequency of a word occurring in an essay divided by the maximum frequency of words; the global weight is the idf, namely, the total number of essays divided by number of essays in which the word occurs. For the IEA system, the term-by-document matrix is constructed by arranging bag-of-words representations of the essays into a matrix. However, it is uncertain how word weights should be assigned: by pure frequency counts or tfidf. Then the matrix is decomposed and essays are projected to a new semantic space, based on algebraic formulas.

2.2 Second Language Assessment and Automated Scoring

Since automated speech scoring occurs in the context of Second Language Assessment (SLA), a brief overview is important to describe its theoretical and practical backgrounds. Theoretical aspects are reviewed, as well as automated scoring for SLA. Following this, the automated essay scoring literature is discussed, since it has implications for speech scoring; finally automated speech scoring , the focus of this study, is investigated.

2.2.1 Theoretical Aspects

2.2.1.1 Overview

Assessment is broadly defined as “the process of collecting information about a given object of interest according to procedures that are systematic and substantively grounded” (Bachman, 2004, p.7). Second-language assessment incorporates characteristics of other assessments but maintains particular second language characteristics. The meaning of SLA can be communicated through an understanding of the models and gleaning of important terms in the field.

SLA researchers have proposed a number of theoretical models of SLA, which are grounded in various perspectives with different foci. The models help people understand the nature of SLA and provide knowledge of this field from different perspectives. Two influential frameworks are Bachman and Palmer’s (1990) framework of communicative language ability and Canale and Swain’s (1980) communicative competence model. The models typically try to represent one core construct in the SLA field and delineate key elements of that construct. Due to their different theoretical motivations and bases, the constructs in the models vary. For example, Bachman’s (1996) model considers the construct of communicative language ability, while Canale

and Swain (1980) studies the construct of communicative competence. While the meaning and key elements of these construct differ, they have in common an attempt to understand the nature of language assessment, as well as to guide test design in practice.

The above-mentioned two models are briefly described here. In Bachman and Palmer's (1996) framework, two factors -- language knowledge and strategic competence -- are important in the construct of communicative language ability of second-language speakers. Language knowledge refers to linguistic discourse knowledge in memory that allows one to compose language output. As in speaking and writing, it can be split into organizational knowledge and pragmatic knowledge. Strategic competence is a set of metacognitive strategies to produce language according to context; strategies include: goal setting, assessment, and planning (Bachman & Palmer, 1996). This model is an altered version from an earlier one in Bachman (1990), which originally contains three factors: language competence, strategic competence, and psychophysiological mechanism.

Canale and Swain's (1980) model depicts the construct of communicative competence. Starting from Chomsky's division between performance and competence, Canale and Swain posit that for Chomsky (1965), "competence" refers to grammatical competence, such as grammatical knowledge of a language. In addition to grammatical competence, Canale and Swain's (1980) have two more types of competence that are considered important: sociolinguistic and strategic competences, because knowledge of language use in social context and communication strategies also matter in communicative competence. Canale (1983) adds one other competence, discourse

competence, to the model, resulting in a four-factor model (Canale, 1983; Bagarić & Djigunović, 2007).

In addition to grammatical competence, important aspects of language competence are sociolinguistic and strategic competences (Canale & Swain, 1980). In assessing speakers, not only knowledge of grammatical rules matters but also knowledge of language use in social context and communication strategies. Thus, Canale and Swain (1980) expand the meaning of competence to the scope of communicative competence to declare three specific sub-types: grammatical competence, sociolinguistic competence, and strategic competence.

As the field of SLA evolves, the models are revised to reflect changes over time in theoretical understandings and empirical results. One important development is the strengthened understanding of key concepts; two of which, language competence and performance, have been discussed extensively because of their central roles in the field. These discussions often involve the definitions of, and the relationships and differences between the two concepts (Chomsky, 1965; Canale & Swain, 1980). Chomsky (1965) proposed differentiating between language competence and performance, or namely between the speaker's linguistic knowledge and his or her actual use of language. Canale and Swain (1980) discuss the difference between communicative competence and communicative performance; competence is about language knowledge whereas performance is about the realization of competence in a particular context. Performance is observed and measured in second-language tests, while competence is not directly measurable because it structures second-language

knowledge (Canale &Swain, 1980). For language testing, the main implication is that that testing performance does not necessarily accurately reflect language competence.

Bachman's (1990) model can serve as a construct foundation for designing language tests, which might have different foci of language ability measurement. It is important to consider the totality of the language ability construct when designing tests (Bachman & Palmer, 1996). The speech construct in this study is also a manifestation of Bachman's (1990) model (Xi, Higgins, Zechner, & Williamson, 2008), of which the measurement of the content-relevance feature in the model is an emphasis because the goal is to facilitate content scoring of speech.

2.2.1.2 Speaking Proficiency

Four aspects are important in second language testing: listening, reading, writing, and speaking (Bachman & Palmer, 1996). Since this study's information-related task lies in automated speech scoring, it is important to review the literature on speaking proficiency. The theoretical models of second language assessment provide an overview of the general domain and depict important components in general. The models can be seen as abstractions of the domain that guide operations of second language testing at conceptual level and meanwhile provide a framework for implementation of actual tests in particular contexts.

Because speaking is a form of language-based communication, assessment of speaking proficiency can also employ the theoretical SLA models. Given a speaking test, the theoretical construct of such a test and speaking proficiency can be established according to the context. Therefore, the construct of speaking proficiency is usually shaped by both the theoretical abstract models and the particular context.

Bachman's model of communicative competence is instantiated in the Test Of English as a Foreign Language, internet-Based Test (TOEFL iBT test) (Xi et al., 2008). As the communicative competence model (Bachman 1990; Bachman & Palmer, 1996) points out key components, the TOEFL iBT speech construct concretizes them in the TOEFL testing environment, which aims to assess English use in the academic context. In the TOEFL iBT speaking framework (as shown in Figure 5), the construct is comprised of three main aspects: delivery, language use, and topic developments. The delivery aspect encompasses acoustic variables like fluency and pronunciation, while the language use aspect includes vocabulary and grammar variables and the topic development aspect focuses on issues such as coherence and content relevance.

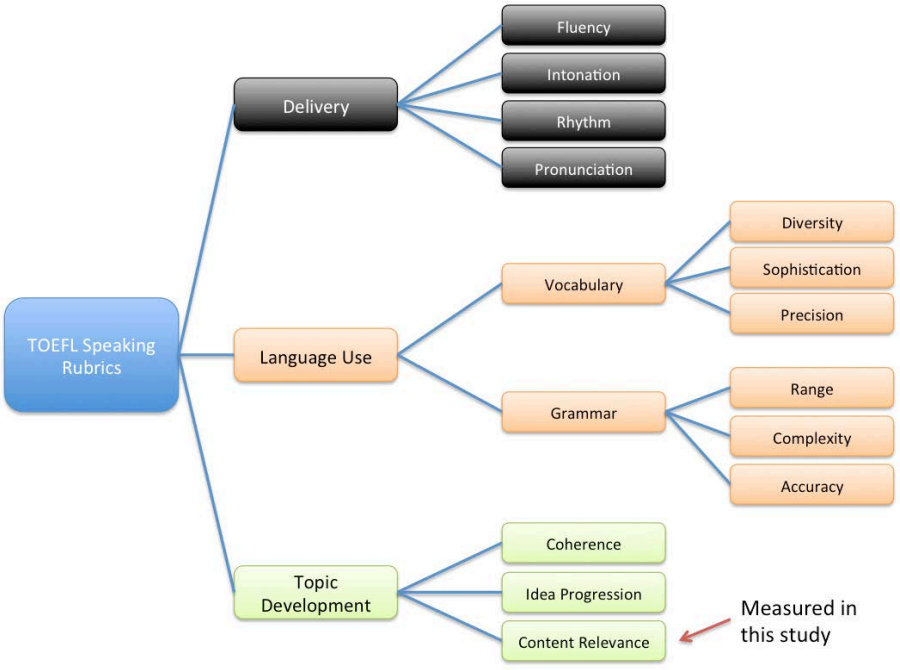


Figure 5. The construct of speech for the TOEFL speaking test (Xi et al., 2008).

This study refers to the TOEFL speech construct (Xi et al., 2008) because the data set is from the speaking section of TOEFL Practice Online tests. The speech construct delineates important features (also called "variables" or "factors"), from which

the author selects topic relevance as a focus because there has been less research on topics than other features. Topic relevance is measured using automatic approaches and then evaluated based on predictive ability of speaking proficiency.

The construct of speaking needs to be theoretically sound and to be validated empirically too. Hence, researchers have undertaken empirical studies to validate the construct in a particular context (Xi et al., 2008; De Jong, Steinel, Florijn, Schoonen, & Hulstijn, 2011), as well as to examine the predicting ability of linguistic features on speaking proficiency (Iwashita, Brown, McNamara, & O'Hagan, 2008; Chen & Zechner, 2011; Cucchiarini, Strik, & Boves, 2000). The speaking features of interest can be extracted either manually or automatically for examination and are typically compared to human-assessed speaking proficiency for validation or prediction examination.

For construct validation, De Jong et al. (2011) conducted experiments to analyze structural components of speaking proficiency. The speaking test design adopts Levelt's (1989) and Levelt, Roelofs, and Meyer's (1999) model of speaking, consisting of: Conceptualizer, Formulator, and Articulator, which are concretized by seven types of measures, including: vocabulary knowledge tests for Conceptualizer, picture naming task for Formulator, and delayed picture naming task for Articulator.

Relationships between measurements and speaking proficiency are analyzed based on experimental results, which show that all are significant parts of the construct, except the two delayed picture naming measures. Xi et al. (2008) collected and analyzed data from TOEFL Online Practice tests to validate the TOEFL speaking construct. The construct's features are realized by lower-level features that can be computationally derived from speech using speech and language processing

techniques. They propose a set of features to be linked to the construct features and implement them in the automated scoring system. The implemented features are also evaluated empirically for the extent to which they cover the speaking construct (Xi et al., 2008).

Experiments and analysis have been conducted to examine the predicting ability of various types of features for speaking proficiency. One feature can be measured in multiple ways; thus, it can be implemented and evaluated based on the different measures. Iwashita et al. (2008) employ a series of linguistic features to investigate their distinguishing power on proficiency levels and find that some vocabulary (token) and fluency (speech rate) features have the strongest impact on proficiency. Cucchiarini, Strik, and Boves' (2002) study compares the relationship between several automatically derived measures and fluency on spontaneous and read speech. Seven automatic measures of fluency are presented and compared to human perception of fluency. The results show that read speech has more measures strongly related to perceived fluency than spontaneous speech, and measures predictive of both types of speech include the rate of speech and the number of silent pauses per minute.

2.2.2 Automated Scoring for Second Language Assessment

Today, more and more second language speakers are taking language tests, creating a great amount of scoring tasks for test agencies. For example, in 2007, IELTS first-time takers exceeded 1 million and in 2009 1.4 million people took the test ("IELTS," 2013). Traditionally, test responses were scored by human graders, but the speed of delivering scores becomes problematic when the number of test takers

increases exponentially. Therefore, there is a need for quick, accurate, and objective scoring delivered to test-takers and automated scoring presents a reasonable solution.

Valenti, Neri, and Cucchiarelli (2003) point out that the benefits of adopting Automated Essay Scoring (AES) systems are: 1) avoiding perceived subjectivity caused by human assessors and consistent scoring; and 2) reducing costs and saving time. These benefits actually can be applied to all automated scoring systems, including speech contexts. Automated scoring systems, which a number of studies have evaluated, meet the needs of the increasing number of people taking language tests. In the next two sections, AES systems are reviewed as they will serve as baselines in this study, and then the current status of automated scoring of speech is surveyed.

2.2.2.1 Automated Essay Scoring

Regarding the technical aspect, AES is a special type of text processing which identifies useful features, extracts them from the essay text, and builds scoring models from them to predict second-language learners' writing proficiency. Since it belongs to the text processing category, various text processing techniques can be employed on essays to achieve this task; on the other hand, when assessing writing proficiency, the applicability of the techniques need to be considered beforehand for particular contexts.

The earliest AES is the PEG system (Project Essay Grader), which uses extrinsic features (proxes) to approximate intrinsic features (trins). The intrinsic quality of an essay is not directly measured but is correlated and predicted by some surficial features. For instance, count of vocabulary (a prox) can be used to predict fluency (a trin) (Dikli, 2006). Other essay scoring system examples are Intelligent Essay Assessor (IEA), e-rater, IntelliMetric, and BETSY (Bayesian Essay Test Scoring System), all of

which rely on natural language processing to derive features from texts and employ various statistical approaches for final scoring. Among them, the mechanisms of e-rater and IEA systems have been selected and used as baselines for this study because of their prevalence in the field.

The workflow of an automated essay scoring system can be divided into two modules: 1) feature identification and extraction; and 2) scoring model construction. First, the author introduces features of essay scoring, which are typically linguistic features that exhibit high correlation to writing proficiency in theory or practice. Features at various linguistic levels can be used to extract features on multiple levels, for example: the discourse, semantic, syntactic, and lexical levels. For instance, syntactic level features can be used to measure the syntactic complexity of essays and lexical level features can be used to measure their vocabulary richness. AES systems have been designed for a variety of perspectives and correspondingly are represented by various feature sets. The features at different levels are usually connected to one or more variables in the language proficiency construct.

The feature set of the e-rater system has been updated as the system evolves from v1.0 to v2.0. It analyzes linguistic features on discourse, syntactic, and vocabulary levels (Valenti et al., 2003). E-rater v1.0 contains features of discourse structure, syntactic structure, and analysis of vocabulary usage (topical analysis) (Burstein, 2003), whereas E-rater v2.0 consists of more features, including: grammar, usage, mechanics and style measures, organization and development, lexical complexity, and prompt-specific vocabulary usage. Another important system, the IEA system, establishes a semantic vector space for essays by using LSA techniques for computing content

features. In addition to content features, IEA contains non-content features such as style and mechanical features (Dikli, 2006).

AES systems also use diverse scoring models that consider features as input and output predicted scores for essays. The scoring model typically follows the training-testing paradigm, which means some essays are used to estimate scoring model and some are used for evaluating the model. A handful of models applicable to the training-test paradigm have been employed in the scoring systems, for instance: linear regression for e-rater and IEA (Burstein, 2003; Landauer et al., 2003), Bayesian classification model in BETSY (Dikli, 2006), and the system delineated in Larkey & Croft (2003).

Before identifying and extracting features, it is important to represent essays in a way that can facilitate processing. This is not always explicit; for example, people might not be aware that they are using bag-of-words representation when they count vocabulary size of an essay. However, feature processing can be significantly affected by representation and features will look different based upon different representation approaches. Representation is the intermediary between essays and features, and the effects of different systems need to be explicitly evaluated.

This study focuses on the content scoring of speech because content scoring of essays is an important subfield. Content features, which are used to facilitate content scoring, are included in several essay scoring systems. In the e-rater system, content similarity between two essays is measured by cosine similarity in the vector space, and content relevance is measured by two features: 1) the level of training essays that the test essay has the largest content similarity to, and 2) the test essay's content similarity

to highest score training essays (Burstein, 2003). For the IEA system, two primary content features are: 1) the essay's quality, as measured by cosine weighted average of the scores of the 10 most similar sample essays and 2) the amount of domain relevant information it contains, which is the vector length of this target essay (Landauer et al, 2003).

Both e-rater and IEA employ cosine similarity for content similarity calculation, although they differ in how to construct the vector space. E-rater uses bag-of-words and IEA uses LSA vectors for essay representation. Consequently, since the representations of essays in vector space are different, the content similarity between essays varies, as do scoring models. Going beyond the two representations, this study proposes using ontology-based representation, which is reviewed in section 2.3.2.

2.2.2.2 Automated Speech Scoring

Given a particular context, the construct of speaking proficiency typically guides both second language speaking tests and the architecture of automated scoring systems for second language speech. In other words, automated scoring systems need to follow the roadmap of the speaking proficiency construct and facilitate automated measuring of the important factors in the construct.

Similar to AES, speech scoring systems extract linguistic features and compute them in a way that measures the factors. For example, speech scoring systems can compute the number of words a speaker says per minute to measure the fluency factor of speaking proficiency. Many automated speech scoring systems adopt this manner of extracting speech features, measuring speaking factors, and building scoring models based upon extracted features to predict speaking proficiency.

The first computerized system for speech scoring was developed by the Ordinate Corporation, for the PhonePass test, to provide linguistic analysis and automated scoring functions (“Versant”, 2013). Over time, its name has been changed to PhonePass SET-10 and then Versant. The Versant framework includes an acoustic model for speech recognition, a dictionary, a language model, and a trained pronunciation and fluency model to score speakers (Dodigovic, 2009). Another influential automated speech scoring systems is the SpeechRater system developed at ETS, which follows the construct of the TOEFL speaking construct (Xi et al., 2008). It consists of three primary modules: speech recognition module, feature computation module, and scoring module (Zechner et al., 2009).

Both Versant and SpeechRater perform quite well on the scoring task. One way to measure performance is by computing the correlation between system predicted scores and human assigned scores, which indicates agreement between the automated and human scoring. Using this measure, correlations for the Versant and SpeechRater are 0.75 and 0.57 respectively (Oridnate, 2005; Xi et al., 2008). This means that the machine decision is relatively close to the human decision, though there is still ample space to improve correlation.

As stated before, speech contains sound and text layers, which is reflected in the speaking proficiency construct. In the TOEFL speaking construct, the fluency factor belongs to the sound side, and the content relevance belongs to the text side. Speech scoring systems integrate the features belonging to the two layers. Generally, the sound (acoustic) features and text features aim to address the construct’s sound and text factors, respectively.

On the sound side, such proficiency factors as fluency, pronunciation, grammatical accuracy, and vocabulary diversity have been implemented in the SpeechRater (Zechner et al., 2009). Features such as pace, fluency, and pronunciation are included in the Versant system ("How the Versant Testing System Works"). One factor can be measured by aggregating one or more features; for example, the fluency factor subsumes nine features and is thus measured by nine features in the SpeechRater. Meanwhile one atom linguistic feature can be the measurement for one or more proficiency factors. For instance, the "tpsec" feature (types per second, types being words) of SpeechRater measures both fluency and vocabulary diversity (Zechner et al., 2009).

So far, features from the text side of speech have been less frequently involved in automated speech scoring systems. As shown in the early SpeechRater system, the only text-related features are vocabulary diversity and grammatical accuracy (Zechner et al., 2009). The Versant system employs such word features as vocabulary as text features (Bernstein et al., 2010) but, again, text features are not heavily involved. More recently, Xie, Evanini, and Zechner (2012) experiment with three content similarity measurements for speech content scoring, including vector space mode, LSA, and pointwise mutual information. The field is evolving with more content features investigated, while more text features still need to be implemented and added to the scoring system to enrich the construct coverage of the systems. This study aims to contribute to this gap in the literature.

2.3 Ontology and its Use in Text Processing

2.3.1 Definitions of Ontology

The term “ontology” comes from philosophy; it originated in ancient times, although its meaning also has evolved over time. Starting from Aristotle, ontology was used to categorize beings of the world (Sowa, 2000). In his masterpiece *Categories*, Aristotle lists ten abstract categories to which all things in the world should belong (Sowa, 2000; Johansson, 2004). From the sixteenth century, the term “ontology” was widely studied as a branch of philosophy and recently, the concept has been used widely in the field of artificial intelligence.

Quine asked the fundamental question of ontology, “what is there?” to which the answer was “everything” (Sowa, 2000). In this sense, it can be construed that ontology studies the nature of existence. Kant proposed four groups of triads to represent all existence, to which Hegel added more triads in follow up (Sowa, 2000; Johansson, 2004). Peirce built upon Kant’s triads, proposing categories of existence: firstness (independent of external relationships), secondness (depend on external relationships), and thirdness (the mediation that connects firstness and secondness)(Sowa, 2000). Sowa (2000) considers these triads to be consistent concepts for Aristotle, Kant, Hegel, and Whitehead, despite different nomenclatures and slightly different meanings.

According to Sowa (2000), the highest level of ontology is the “thing,” which has no properties and covers all existence. The lower-level categories are based upon different properties, which according to Aristotle are: quantity, substance, quality, and seven other categories (Sowa, 2000; Johansson, 2004). In fact, an ontology can be extracted from every philosopher’s work (Sowa, 2000), since it is about his or her categorization of being.

After the development of artificial intelligence (AI) in the 1960s, ontology was imported from philosophy by the AI and other information-related communities to facilitate research. Ontology can help with the knowledge representation tasks, since it categorizes existence based on properties, in a general sense. A central issue in AI is studying approaches to representation of knowledge for computation. Meanwhile, ontologies in AI are strongly tied to computer processing because its goal is to utilize theories of ontologies to represent knowledge in a machine-understandable manner.

In AI, ontology inherits some characteristics of the philosophical definition, although it addresses reality of a specific domain rather than the whole world and contains more detailed knowledge of that domain. Another difference from philosophical ontology is that AI includes characteristics to make ontology practical and implementable, so as to facilitate information-system applications. The resulting outcome is that definitions of AI ontologies frequently discuss components of ontologies, in order to be processed by computers.

One widely cited definition of AI ontology is "*ontology is a specification of a* conceptualization" by Gruber (1993). Furthermore, Gruber provides an explanation of his definition:

"An ontology is a description (like a formal specification of a program) of the concepts and relationships that can exist for an agent or a community of agents (Gruber, What is an Ontology?)."

Conceptualization can be understood as modeling, and Gruber's second definition illustrates the specification of conceptualization in a more practical way. That is to say, first, ontologies need to have concepts and relations as components. Second,

these components should be agreed upon by a group of people, or a “community of agents.”

Guarino (1998) provides another widely cited definition of conceptualization, which is an integral part of ontology:

“Consider a logical language L using a certain set V of predicate symbols, called the vocabulary (or the signature) of that language. When an agent A uses L for some purpose, the intended models of L according to A will constitute a small subset of the set M(L) of all models of L. We call such set of intended models the conceptualization of L according to A.”

This definition is similar to Sowa’s in that they both claim that a language is needed for representation. But Gruber emphasizes modeling, whereas Sowa talks about categorizing; these authors discuss ontologies of different fields. The agreement on ontological elements among agents is explicitly mentioned in Gruber’s (1993) definition, while in Guarino’s (1998), the word “agreement” is not used but the word “agent” implies the meaning of a group of people too.

To represent knowledge in a computable way, one needs to write ontology in a formalized language, which facilitates computer understanding of knowledge, as well as the interaction between humans and computers. Usually, the elements of ontologies are: concepts, relations, properties, and axioms. A number of ontology definitions focus on pointing out the elements, as illustrated and summarized below.

Noy and McGuinness’s (2001) definition suggests that ontologies have “*concepts in a domain of discourse (classes (sometimes called concepts)), properties of each concept describing various features and attributes of the concept (slots (sometimes called roles or properties)), and restrictions on slots (facets (sometimes called role restrictions))*.” Thus, ontology elements should be: concepts, properties, and restrictions

on properties. Gruber (2009) decomposes ontologies into classes, properties, and relationships. For Neches et al. (1991), ontologies have elements of: basic terms, relations, and rules. Swartout, Patil, Knight, and Russ (1996) show that an ontology is a "*set of structure terms*," meaning that terms and structures together constitute ontologies.

Summarizing elements from the above definitions, ontologies typically contain concepts, properties, relations, and rules. Moreover, for comprehensive representation of domain knowledge, they also contain such elements as instances and axioms. Of all the elements, concepts and relations are the most frequently used in existing ontologies and can be seen as the building blocks of a comprehensive ontology.

Concepts, which usually describe abstract and conceptual terms, as compared to instances, which are about entities in the real world, are the most basic elements in ontologies and can be called classes, objects, or entities. For example, "car" is a concept since it is a collection of a specific type of entity, and a specific car in the real world is an instance. They may vary in terms of task, function, action, strategy, reasoning process, and other ontological aspects (Gómez-Pérez & Corcho, 2002).

Properties are also called attributes, which are the features or characteristics that make concepts group together, as well as what distinguishes one from another. Every concept needs properties in order to be grouped somewhere and to be distinguished from other concepts. For concepts organized in a hierarchy, concepts in lower levels can usually inherit properties of concepts in higher ones, and in the meantime, particular lower-level concepts have their own properties that differ from other lower-level concepts.

Two general ontology examples are WordNet, and Wikipedia. They are considered as ontologies in this study because their components align well with critical components of ontologies. WordNet is a lightweight ontology, which encompasses concepts and hierarchical relations in the general domain, and which is edited by experts. Wikipedia is contributed by community users and contains concepts, instances, and semantic relations in the general domain. In addition, domain specific ontology examples include the UMLS ontology that describes knowledge in the biomedical domain, containing terms and semantic relations contributed by experts. Because WordNet, and Wikipedia are the ontologies used in this study, more details of each are briefly presented below.

WordNet and Wikipedia

WordNet is a computational lexicon of English, with words as its basic units (Fellbaum, 1998, p.4). As one word may have several meanings (senses), senses of words are analyzed and similar word senses are grouped to a same concept, called “synset” in WordNet. The synsets, which subsume one or more synonymous words, can be thought as ontological classes or concepts (Hotho et al., 2003a). The current version, v3.0, contains 117,659 synsets (“WordNet Statistics,” n.d.). WordNet synsets are also connected via semantic relations, usually as hierarchical relations such as is-a and part-whole relations, to form a semantic network. WordNet encompasses four types of words: nouns, verbs, adjectives, and adverbs, of which nouns and verbs are hierarchically structured by is-a relations (“WordNet,” 2013). The semantic relations between synsets can be treated as relations and thus synsets and relations together form the WordNet ontology.

Wikipedia is a collaboratively edited web encyclopedia contributed to by a community of people (“Wikipedia,” 2013). It is a knowledge base covering vast topics with relatively high editing quality. Furthermore, it can be thought of an ontology because it aligns well with the essence of ontologies in three aspects: 1) it delineates knowledge of the general domain, 2) article titles (or topics) can be considered as ontology concepts, and 3) the Infobox is a fixed format table containing metadata of a topic, and its metadata fields can be seen as properties of a concept, 4) some Infobox metadata fields navigate to another page and thus form a relationship between two articles, and this can be thought as relations between concepts, and 5) category information of a topic can be seen as the parent concept of this topic, similar to hypernym in WordNet. The table below shows the mapping between WordNet/Wikipedia components and ontology components.

Table 1. Mappings between WordNet/Wikipedia and ontology components.

Wordnet components	**Ontology components**
Terms (noun, verb, adjective, adverb)	Concepts
is-a relation	Semantic relations (hierarchical)
Wikipedia components	**Ontology components**
Articles	Concepts
Infobox metadata	Properties
Navigation links	Semantic relations
Category information	Semantic relations (hierarchical)

WordNet can be more easily mapped to an ontology because the terms have clear hierarchical relations between them. In contrast to the strictly structured WordNet, Wikipedia is loosely structured in a hierarchy organized by Wikipedia categories;

however, it has larger coverage of concepts, metadata for concepts (properties), and potentially more types of concept links than WordNet. Except for concepts, other ontology components such as properties and relations are not well formalized in current Wikipedia. That is to say, though conceptually Wikipedia components align with ontology components, e.g. metadata fields match to ontology properties and navigation links match to ontology relations, if Wikipedia is to be used as a formal ontology in practice, work needs to be done on converting the current Wikipedia version to some ontology languages. For example, on the article page of “Peach”, which has kingdom of “Plantae” as shown in its Infobx, and a semantic relation is established by this piece of Infobox information: concept “Peach” has kingdom of concept “Plantae”. This relation exists in an implicit way that needs to be extracted and described explicitly. Such efforts have been demonstrated in the DBpedia project, which extracts structured information from Wikipedia and arranges the information in an ontology (Bizer et al., 2009).

In summary, firstly, ontology seeks approaches to knowledge representation and suggests implementable formality for knowledge representation. These approaches enable other research areas to reuse and share knowledge. There has been much research into methodologies of building ontologies, such as the top-down (Gruber, 1993; Gruninger & Fox, 1995; Swartout, 1996) and the bottom-up approaches (Bisson, Nedelee, & Canamero, 2000; Mani, Samuel, Concepcion, & Vogel, 2004; Schmitz, 2006). For better share and reuse, ontologies can be formalized in ontology languages, for example, in semantic web languages such as RDF (Resource Description Framework), RDFS (RDF Schema), and OWL (Web Ontology Language). Regardless of the language it is written in, a knowledge base can be identified as an ontology if they

follow the generally accepted definition of ontologies. For example, the author recognizes WordNet and Wikipedia as ontologies since they match well to ontology essentials such as Gruber's (1993) definition, and they are to be used for text processing in this study.

Secondly, a relevant question is how ontology can be used to facilitate information-related tasks, given the existence of so many ontologies. A specific focus of this study is how ontology can facilitate representing speech transcripts and further content scoring of speech. Ontologies provide rich concepts and relations in general or specific domains, which are useful in understanding natural language text in relevant domains. More specifically, ontologies contain structured knowledge, whereas text documents are unstructured; therefore, the structured knowledge of ontologies may assist the uncovering of concepts and relations in the unstructured documents and thus facilitate further computational analysis of the documents. In fact, ontologies have been utilized in text processing in numerous studies, from which the approaches are presented in the next section.

2.3.2 Use in Text Processing

This study involves text representation and reasoning about the importance of unknown concepts, while this section focuses on using ontologies in text representation and for concept similarity measurement. Ontologies have been used for text processing tasks in representation, such as text classification and clustering. Concept similarity measurement will also assist with unknown concept reasoning by assessing similarity between unknown and known concepts.

Ontology-based representation in text processing can be referred and applied to representation of speech transcripts. Sebastiani (2002) summarized several assumptions underlying text categorization using machine learning techniques. One assumption addresses sources of knowledge, which is said to come from endogenous knowledge from corpuses with no exogenous knowledge. The use of machine learning in text categorization makes the learning process lack external knowledge and thus is a shortcoming of this approach.

Bloehdorn and Hotho (2004) mention that the primary features of text categorization have been bag-of-words, indicating the representation in text categorization can inherit shortcomings from bag-of-words. It is to some extent a reflection of Sebastiani's (2002) assumption of lack of exogenous knowledge, which also applies to text clustering tasks, which in turn share the same shortcomings from document representation with text categorization. In fact, a number of studies have gone beyond this assumption and employed such knowledge-based approaches as ontology to complement the lack of exogenous knowledge in machine-learning methods (Bloehdorn & Hotho, 2004; Hotho et al., 2003a; Hotho et al., 2003b; Zhang, 2009; Gabrilovich & Markovitch, 2007).

Hotho and Bloehdorn, along with others, conducted a series of studies using ontologies for text categorization and clustering tasks (Bloehdorn & Hotho, 2004; Hotho et al., 2003a; Hotho et al., 2003b). The goals are to overcome several weaknesses, like synonym and generalization issues, of the bag-of-words representation by using ontology concept based representation. Basically, concepts from ontologies are used as units for text representation and text processing is performed on top of the ontology-

based representation. The ontological concepts construct a semantic vector space, as opposed to the word vector space in the bag-of-words representation. One benefit of the ontology-based representation is that synonyms are grouped in the same dimensions; another is that higher-level concepts can be used in the representation to unravel semantic relations between documents containing the same higher-level concepts (Bloehdorn & Hotho, 2004; Hotho et al., 2003b).

Bloehdorn and Hotho's (2004) study focuses on the text categorization task, and Hotho et al. (2003a; 2003b) work on text-clustering tasks. The preprocessing steps all include locating concepts in the text by matching text to ontology. The difference is that Bloehdorn and Hotho (2004) match the maximum string in text to ontologies to find the most specific concept, whereas the other two studies match single words to concepts in ontologies.

The three studies employ and experiment with the same parameters: 1) asking whether concept features should be used alone or to replace word features, or should be used together with word features; 2) word sense disambiguation strategies when using concepts, options including determining word sense based on its 1st WordNet sense, part-of-speech role, and context; 3) levels of concept generalization in ontologies, namely, how many levels to go up to trace higher-level concepts and use them to expand the representation (Bloehdorn & Hotho, 2004; Hotho et al., 2003a; Hotho et al., 2003b).

The studies observed positive results in using ontology-based representation, with the best results on several corpora occurring in the parameter setup, which uses both concept and word features in representation, performs word sense disambiguation

based on context, and traces up 5 levels higher to include the all other concepts between the concepts themselves and parents 5-levels up in the representation vector (Bloehdorn & Hotho, 2004; Hotho et al., 2003a; Hotho et al., 2003b). Due to their good performance, some of these ontology-based representation approaches will be utilized in this study's experimental design.

Zhang's (2009) dissertation study explored methods of using knowledge sources to enhance text mining. In this methodology, text was matched to ontology concepts in preprocessing in a similar manner to the previously discussed studies. In one sub-study, the 5-gram of PubMed articles is matched to MeSH (Medical Subject Headings) concepts. In another sub-study, documents were mapped to Wikipedia concepts in two ways: exact matching, which directly identifies Wikipedia concepts in documents; and relatedness matching, which first represents words by vectors of Wikipedia concepts by using text description of these concepts and then represents a document by vector of Wikipedia concepts based on the words it contains, given that the representation of words by vector of Wikipedia concepts is known. For text clustering tasks, after documents are represented by vector of concepts, document similarity can be computed by cosine similarity between document vectors (Zhang, 2009; Hotho et al., 2003b).

The Explicit Semantic Approach (ESA), as proposed by Gabrilovich and Markovitch (2007), represents an arbitrary text snippet in a vector of Wikipedia concepts for the purpose of natural language processing. Each Wikipedia concept has a text description, which is used to build an inverted index to associate words with concepts. The invert index helps represent each word by a vector of other Wikipedia concepts,

and eventually a document can be represented by weighted Wikipedia concepts by adding up the concept vectors of the words contained in the document.

The above ontology-based representations largely take advantage of the concepts in ontologies, while semantic relations, an important part of ontologies, are also utilized in text processing. Semantic relations are often used to measure semantic similarity and distance between objects like words, phrases, named entities, concepts, and documents. The connections between ontological concepts play an important role in concept similarity measurement.

WordNet and Wikipedia are two popular ontologies for computing semantic similarity. A number of similarity approaches have been proposed for similarity calculation according to the different characteristics of the two ontologies. WordNet is tree-structured with hierarchical relations between concepts, while Wikipedia is graph-structured, with both hierarchical and non-hierarchical relations among concepts.

Relying on the WordNet IS-A structure, edge-based similarity and information content-based similarity can be employed to compute concept similarity. Edge-based similarity uses path information between two hierarchical concepts; three examples are *path*, *lch*, and *wup* (Pedersen et al., 2004). Path similarity uses the shortest path between two concepts; lch scales the shortest path by the maximum path length in the hierarchy, and wup finds the most specific ancestor subsuming the two concepts and counts the path from the ancestor to the root node (Pedersen et al., 2004). The information content-based similarity makes use of external corpus in addition to WordNet structure. Resnik (1999) proposed using information shared by the two concepts to measure similarity by examining the information amount of the least

subsuming concept. The paper claims that the least subsuming concept is more informative if the two concepts are similar and less informative if they are dissimilar. The amount of information of the least subsuming concept can be computed by the probability of the concept occurring in an external corpus, such as the Brown corpus; the derived logarithm of the probability stands for the similarity between the two concepts (Resnik, 1999). Lin's (1998) similarity is another information content-based similarity, which scales the probability of the least subsuming concept in Resnik (1999) by the sum of the probability of the two concepts. Compared to edge-based similarity, information content-based similarity can overcome the unreliability of paths in the hierarchy (Resnik, 1999).

To compute similarities between Wikipedia concepts, several approaches can be employed. Strube and Ponzetto (2006) tailored similarity measurement for concept relatedness in IS-A taxonomy to the characteristics of Wikipedia. The category tree of Wikipedia is treated as taxonomy and edge-based similarity like lch, wup, and Resnik's (1999) information content similarity are employed on Wikipedia concept pairs. The rich links of Wikipedia objects provides basis for computing semantic relatedness.

Milne and Witten (2008) proposed two measurements to employ Wikipedia hyperlinks for concept relatedness: one represents Wikipedia articles using vectors of outgoing links and calculates the cosine similarity between the vectors of links as semantic relatedness; the other, which is similar to the Google similarity (Cilibrasi & Vitanyi, 2007), uses Wikpedia articles linking to target concepts to compute relatedness.

2.4 Summary

The author has reviewed three important research areas to be addressed by this study, including document representation, second language assessment, and ontology and its use in text processing. The document representation review provides a basis for the baseline representation approaches in the experiment design. The second language assessment section presents the status of automatic scoring from its theoretical to practical aspects, which reveals the literature gap in the field. The baseline systems in experiment design refer to the representation and feature computation approaches in current essay scoring systems.

Ontology-based representation is the design for experimental approach and existing methods of using ontologies in text processing can be referred to when representing the content of speech transcripts. In sum, this review presents grounds for speech scoring and develops relevant approaches for experiment design.

CHAPTER 3. METHODOLOGY

3.1 Overview

This study uses experiments as the methodology. Based on the literature survey along with theoretical analysis and the nature of ontologies and speech transcripts, the author proposes to use ontology-based approaches to representing speech transcripts in addition to classical representations. In this empirical study, the details of the approaches are delineated, experiments are designed to collect empirical evidence, and analysis is performed to evaluate effects of the proposed approaches in the automated speech-scoring task.

This study considers only features for content representation among many possible ones in speech-scoring models. It does not take into account prevalent features of speech such as fluency, pronunciation, and prosody, as the focus is on the content aspect of speech. As a result the scoring models comprise only features from content. Due to this focus, the experiments will be conducted on speech transcripts produced by human transcribers. In the context of this study, the features[2] extracted from text for vector representation are defined as “features from content” (in machine learning) while the variables and factors related to content in a construct are used as “content features” (in automatic scoring).

The experimental design follows the experiment and control fashion, in which the proposed approaches belong to the experiment group and the baseline approaches

[2] “Features” here are in the sense of machine learning, e.g. high dimensional content vector extracted from a text document. In literature of automatic scoring features usually refers to something different, namely, variables or factors of a construct. For instance, content feature means content relevance of a document; moreover, it can be computed based on content vectors of documents and this demonstrates the connection between machine learning features and features in automatic scoring.

belong to the control group. The baseline approaches manifest the typical practice in the field of automated scoring and stand as the comparison basis for the experiment group. The experiment group encompasses two ontology-based approaches; for the control group, two prevalent systems in automated essay scoring are identified as the baseline systems. Essay scoring systems are used as baselines because current automated speech-scoring systems seldom contain content features. The result from using it in speech scoring may also have significant implications for essay content scoring. Another reason for using essay-scoring systems as the baselines is that they deal with written text that is similar to speech transcripts although they do differ in some ways.

The experiment and control groups are both about content representation of the same speech transcripts, which further leads to building scoring models and predicting speaking proficiency. Each experiment adopts the training-testing data partition for model building and evaluation. The effects of the representation approaches are evaluated based on the performance of the scoring models. Effect analysis and comparison is performed on the approaches of different groups as well as approaches in the same group to present a comprehensive picture of the performances of the approaches.

All the baseline and proposed approaches follow two-step processes: the representation process and the machine learning process, which are the two modules prior to evaluation as presented in section 1.5.2. The baseline and proposed approaches differ in the representation process while sharing the same machine learning process. This is for the convenience of comparing representation approaches,

because performance differences can better reflect different effects of representations with machine learning process set-up remains the same.

For the baseline systems, two prevalent systems in essay scoring, e-rater and Intelligent Essay Assessor, are deployed as the control group of representation approaches. The e-rator system employs the Bag-Of-Words (BOW) representation while the Intelligent Essay Assessor uses Latent Semantic Analysis (LSA) representation. Both of them belong to the statistical representation category.

The experimental systems intend to represent transcripts at the concept level by using ontologies. The ontology-based representations utilize concepts and relations in ontologies to help resolve the challenges of the statistical-based approaches mentioned in section 1.3: meaningfulness of the representation and unknown terms. The proposed representations address the two challenges of the statistical representation approaches. The first approach, ONTO, is to tackle the problem of the meaningfulness of representation issue by using concept level representation; and the second approach, OntoReason, is to address the unknown term problem by reasoning unknown terms based on ontology semantics.

The outputs of representation approaches are ingested by machine learning models as inputs. All representation outputs are processed by the same machine-learning model regardless the approach used and evaluated by the same performance measurement to quantitatively compare between the representation approaches. The machine learning algorithm first builds scoring models from training data and then evaluates model performance using test data. The machine-learning model performance indicates its predictiveness on speaking proficiency given the

representation. Therefore under the same machine learning model with everything set up the same, machine learning performance can be an indicator of the effect of representation approaches because performance differences can be attributed to the difference in representation approaches. This is the logic that the author uses for evaluation of representation approaches.

In the subsequent sections, the data set used in the experiment is introduced in 3.2, then several hypotheses guiding the experimental design are presented in 3.3; sections 3.4 and 3.5 describe the baseline approaches and ontology-based approaches respectively; in section 3.6 the machine learning model for generating scoring models are introduced; and in the last section (3.7), the means of evaluating features and effects of representation approaches are addressed.

3.2 Data Set

3.2.1 TOEFL Practice Online (TPO) data

The data set, the collection of the information resources of this study, was part of the speaking section of TOEFL Practice Online (TPO) test 2006. TPO is an online system where TOEFL Internet-based Test (iBT) test takers can practice and prepare their language tests (Xi et al., 2008). For the TPO speaking test, test takers were asked to provide spontaneous speech responses to the prompts (test tasks). The responses were scored holistically by human raters based on the TOEFL iBT scoring rubric on a scale of 1 to 4, 4 being the highest score (Zechner et al., 2009). For each score level, the rubric lists what performance is expected for speaking aspects such as fluency, pronunciation, and content and guides human raters on assigning a holistic final score based on performance from different aspects.

3.2.2 Prompts

In the context of the TPO speaking test, prompts are test tasks given to test takers to elicit speaking responses. There are two types of tasks used in TPO: independent tasks and integrated tasks. Independent tasks examine speaking ability independent of reading and listening abilities by asking test takers to speak about a familiar topic without giving them reading or listening tasks beforehand; whereas the integrated task evaluates speaking ability along with reading and listening abilities, by giving out test questions after assigning test takers a paragraph to read or an audio to listen to (Xi et al., 2008). The speech responses are spontaneous because their content is difficult to predict.

The data set of this study is a subset of the TPO 2006 speaking responses provided by ETS (Educational Testing Service), a non-profit testing agency specializing in large scale standardized tests. The speech files are in response to 4 TPO prompts, which are coded as prompts 098, 099, 100, and 101 here. The 4 prompts belong to the integrated task category and their information is briefly described in Table 2 (see Appendix B for full content). Test takers provide one response per prompt, with each response being about one minute in length. In the data set, one speaker may provide one or multiple responses.

Table 2. Information of the 4 TPO prompts used in the study.

Prompt Name	Task Type	Reading/ Listening materials	Topic	Speaking task
098	integrated	Reading university president's announcement; listening to discussion between two students.	A university plans to increase tuition and fees.	Test takers are expected to provide opinions based on the materials.
099	integrated	Reading document about animal domestication; listening to part of a lecture on domestication.	Animal domestication.	Test takers need to explain the suitability of antelopes and houses for domestication.
100	integrated	Listening to conversation between two students.	A woman was stressed by her schoolwork and a man suggested two solutions.	Test takers need to talk about the woman's problem and the man's solutions, and then provide their own opinions on the issue.
101	integrated	Listening to a talk about US history.	Automobile and radio helps shape the common culture of US.	Test takers need to speak about the topic using the points and examples in the talk.

3.2.3 Speaking Responses and Data Partition

The data set contains 1237 speech samples in total, which are initially in audio format. Each response was verbatim transcribed by a human, which results in 1237 text files. The transcripts are approximately 121 words in length on average.

Since the main purpose was to examine effects of representations on different prompts instead of having them mixed in one large set, responses were split by prompts, resulting in four distinct groups of responses (Table 3). As shown in Table 3, score 1 category has only a few responses for each prompt (4, 7, 4, 8 respectively). Because it is extremely difficult to train a good classification model for this score level given this number of responses, score 1 and score 2 responses were merged together into "score 2" (Table 4).

The merged data set contains 3 score levels, namely, score 2, score 3, and score 4. Within each prompt group, the responses were further split for the purpose of

cross validation, a standard way of evaluation in machine learning. A 3-fold cross validation was chosen in this study due to the small size of the data set. Each prompt group was split into 3 folds via random stratified sampling to ensure that the scores were distributed in similar proportions to their distributions in the whole corpus. Information of the new data set is in Table 4.

Table 3. Size of original data set (obsolete, not used in experiments).

Prompt	**Score 1**	**Score 2**	**Score 3**	**Score 4**	**Total**
098	4	79	157	78	318
099	7	86	144	69	306
100	4	74	152	79	309
101	8	75	140	81	304
Total	23	314	593	307	1237

Table 4. Size of merged data set (this is the data set used in experiments).

Prompt	**Score 2**	**Score 3**	**Score 4**	**Total**
098	83	157	78	318
099	93	144	69	306
100	78	152	79	309
101	83	140	81	304
Total	337	593	307	1237

3.3 Hypotheses

Four hypotheses were to be tested within the framework of the research questions and used to guide the execution of experiments. As mentioned above, the criterion of measuring effectiveness of a representation approach is the performance of content scoring models computed from machine learning. Hypothesis 1 was formulated as follows to compare the effectiveness of the two baseline systems:

H1*. Content scoring models from LSA representation outperform content scoring models from BOW representation in predicting speaking proficiency.*

The ONTO approach employs ontology concepts as vector dimensions, the effect of which is compared to the BOW baseline in Hypothesis 2:

***H2**. Content scoring models from ONTO representation outperform content scoring models from BOW representation in predicting speaking proficiency.*

Hypothesis 3 was formulated to compare the ONTO approach against the LSA baseline to acquire a comprehensive understanding of its effects:

***H3.** Content scoring models from ONTO representation outperform content scoring models from LSA representation in predicting speaking proficiency.*

The OntoReason approach is built on top of the ONTO approach, and its effect is compared to the ONTO approach in Hypothesis 4:

***H4**. Content scoring models from OntoReason representation have better predictiveness on speaking proficiency than the content scoring models from ONTO representation.*

Figure 6 presents the hypotheses and comparisons between approaches:

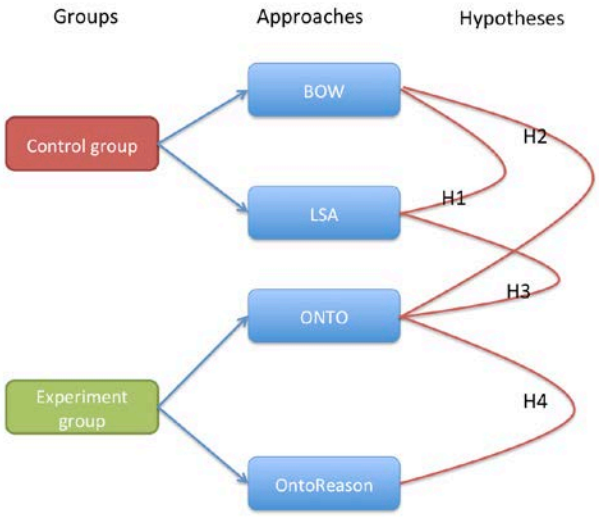


Figure 6. Hypotheses and comparison between approaches.

3.4 Baseline Systems

Each baseline and experimental system was primarily composed of two parts: representation approach and machine learning model. The representation approach was of most importance to the study and presented respectively for each system. As machine learning was used mainly as a tool for evaluating the effects of representations in this study, the same machine-learning model was applied in all baseline experiment systems to control the condition of comparison and maintain the comparability among the evaluation results.

3.4.1 Bag-Of-Words Approach (BOW)

E-rater is an essay scoring system employing the BOW document representation. The following sections discus the e-rater workflow and delineates details of the BOW representation for this study, while not necessarily following every parameter setup of e-rater.

3.4.1.1 Representation

The BOW approach takes the view that essays can be represented in a vector of words and the value of a word in a vector refers to its weighting on this dimension. The vector space construction follows the common practice by using all the words in the documents. The vector weighting is based upon word frequencies in a document along with other information. Given a document, its vocabulary can be extracted and tokens in the vocabulary are used as dimensions of vector representation:

$$d \rightarrow (t_1, t_2, t_3, \ldots, t_n)$$

where d is the document, t_i (i=1,2,…,n) is a token in the vocabulary, and n is the size of the vocabulary.

The author considers weighting method as the parameter for BOW, which estimates the importance of words. For stopwords, she follows the standard practice in the information retrieval field and removes stopwords from transcripts.

Before introducing the weighting parameter, it is worthy pointing out that, although stopwords, or function words, often do not contain as much content meanings as topic words do, they can be important in the vector representation. For examples, negation words (i.e. “no” and “not”) are important in sentiment analysis and need to be retained in the vector or constructing negation-aware representation (Pang & Lee, 2008). In this study stopwords were kept and applied to the text before the BOW vector representation.

Weighting

There are various ways to compute the weights of words in a document vector. This study chose 2 weighting schemes and selected the best option from experimental results.

1) Normalized tfidf**.** This is a typical tfidf weighting scheme, which obtains tfidf weights for words in a document and then normalizes the tfidf weights by Euclidean length of the document:

$$W_i = \frac{F_i * \log\left(\frac{N}{1+N_i}\right)}{\sqrt{\sum F_i * \log\left(\frac{N}{1+N_i}\right)^2}}$$

where F_i is the frequency of word i in the document, N is the total number of documents in the collection, and N_i is the number of documents containing word i; the denominator calculates Euclidean length of a document, by summing squares of weights of all the vector dimensions and obtaining its square root.

2) Normalized tf. This weighting scheme contains only term frequencies without multiplying by idf. Since the computation of idf is prompt specific, important topic words may result in low idf values due to their frequent occurrence in the prompt corpus. However, we would like these terms to retain a high weight in vectors, for example, “tuition” should have high weighting values in prompt 098 but its idf is 0.1168, ranked 1213 in 1221 words in the prompt 098 corpus. The underestimated idf values may further lead to underestimation of weights of such topical words on documents. An attempt was made to rule out the influence of idf on important topic words by using only tf for term weighting. Similar to normalized tfidf, tf is also normalized by the Euclidean length of the document:

$W_i = \frac{F_i}{\sqrt{\sum {F_i}^2}}$ where F_i is the frequency of word i in the document.

3.4.1.2 Parameters to be Tuned

The weighting method (with 2 options) needs to be tuned in the BOW approach. The author identifies the best weighting option from experiment results. The selected weighting method is not only used in BOW representation, but it will also be used in other approaches for weighting Wordnet synsets and Wikipedia concepts.

3.4.1.3 Implementation Details

For stoplist, a list of 160 stopwords was adopted, which was developed at Educational Testing Service based on the stoplist of the SMART system.

The idf values were prompt specific, meaning the idf values were computed within each prompt instead of the whole corpus. For more precise idf computation, the author computed idf values for each run in cross validation. Responses under each

prompt were partitioned into 3 folds (fold 1, fold 2, and fold 3) to achieve a rigorous machine learning evaluation. In each run, 2 folds were used for training and the remaining fold was used for testing, and there were 3 machine learning runs for each prompt in total. Within each run, the author built an idf list from the 2 training folds for the run, and therefore the idf lists varied in each run. The author did not build a global idf list from the whole corpus in order to keep the test set intact.

3.4.2 Latent Semantic Analysis Approach (LSA)

Latent Semantic Analysis (LSA) is a statistical technology to identify latent concepts in documents and construct a latent semantic space for the documents. The technical details of the LSA approach have been introduced in section 2.1. Basically, it decomposes a term-by-document matrix into three sub-matrices, which will form a latent semantic space for project documents and terms. In this latent semantic space, dimensions can be considered as latent concepts that are used to represent the documents and terms.

3.4.2.1 Representation

A typical use of LSA in essay scoring is demonstrated by the Intelligent Essay Assessor (IEA) system. It represents essays in a semantic space by using LSA decomposition and helps decide the goodness of conceptual semantics of essays (Landauer et al., 2003). The assumption is that “the meaning of a passage is the sum of the meanings of its words” (Landauer et al., 2003). Several publications have described methods of using LSA for essay scoring in the IEA system (Landauer et al., 1997; Landauer et al., 2003; Foltz et al., 1999), while the methods vary slightly from each

other but still share a good amount of commonality. The most consistent parts are described below and adjusted to some extent to form the second baseline system.

LSA generates matrices for specific domains, which needs text sources for deriving latent concepts for the domain. For essay scoring three types of text sources can be used for LSA training: 1) pre-scored sample essays written by students; 2) essays by domain experts or knowledge source materials; 3) internal comparison of an unscored set of essays (Landauer et al., 2003). Landauer et al. (2003) used all the three text sources for LSA-based analysis in scoring essays about heart studies and reported that reliabilities of the LSA-based model using different text sources are comparable to each other. The first type of text sources, pre-scored sample essays, was selected to be the training corpus for LSA. LSA matrices were generated for each prompt as well because of the prompt-specific nature of the representations in the study.

Within each prompt, LSA vector space was generated from the training set. It resulted in three matrices. The three sub-matrices were reduced to *k* columns by designating the rank approximation for dimensionality reduction purpose. Seven values for the ranking parameter *k* (10, 20, 30, 40, 50, 100, 200) were tuned for best fit of the model.

After producing the vector of latent concepts, a vector of concepts for a given test document was computed by the three resulted sub-matrices by this formula: $e^T T_k S_k^{-1}$, where $e^T$was the transpose vector of the document vector (in words), T_k wass the left matrix from the SVD (singular vector decomposition) computation with its first k columns remained, and $S_k^{-1}$was the inverse matrix of S_k, which was the diagonal matrix from SVD with its first *k* values kept.

3.4.2.2 Parameters to be Tuned

The weighting method for composing the original term-by-document followed the best option derived from the BOW approach. Stopwords were removed from transcripts prior to generating term-by-document matrix. The parameter to be decided was the rank *k* in dimensionality reduction of the sub-matrices. The SVD process returned three sub-matrices, and the reduced three sub-matrices based on integer *k* constructed the rank *k* approximation of the original term-by-document matrix. The *k* values (*k*=10, 20, 30, 40, 50, 100, 200) were experimented and the value that generated the best performance was the optimized parameter *k* for the LSA approach.

3.4.2.3 Implementation Details

The Gensim, a semantic modeling package written in Python, was used for the matrix decomposition and latent vector computation tasks in LSA (Řehůřek & Sojka, 2010).

3.5 Experimental Systems

This section discusses two ontology-based representation approaches in detail. The experiment followed the vector representation style and employed the same machine learning models similar to what was done in the two baseline approaches.

3.5.1 Ontology-based Representation (ONTO)

Instead of representing text by words from corpus, text was represented by concepts from external ontologies, Wikipedia and WordNet. The first step in ontology-based representation was to identify concepts from document strings. In this step the document text is usually segmented to substrings such as words and phrases, which are further used to match ontology concepts. In practice, the concept matching method

varies according to the characteristics of ontologies. Below is the description of the representations using WordNet and Wikipedia respectively.

3.5.1.1 ONTO-WordNet

3.5.1.1.1 Concepts in WordNet

Synsets, namely groups of synonyms, are concepts in WordNet. A word may belong to multiple synsets in WordNet depending on its senses. For example, the word "travel" has 9 senses in WordNet and thus belongs to 9 different synsets, such as *{travel.n.01}* and *{change_of_location.n.02}*. Since WordNet mostly contains words, the documents are broken into tokens, which are then matched to WordNet synsets. For concept matching, the author adopted the strategies below for selecting the appropriate synset for a given word.

3.5.1.1.2 Concept Matching Options

In the preprocessing of the documents, the author did not use stoplists because the full context was important for detecting word senses. Document text was split by whitespace and punctuations to a set of words, each of which were then matched to a WordNet synset if such a match existed. Word Sense Disambiguation (WSD) and vector construction were two parameters to be considered in concept matching, according to Hotho et al.'s (2003a).

As a word may have multiple synsets in WordNet, it is important to disambiguate the senses of words to locate the most appropriate synset. Hotho et al. (2003a) propose three ways of conducting WSD for synset matching when given a word in document. Two of them experimented in this study were defined as "1st sense" and "POS" respectively.

1) 1st sense: the synset of a word in WordNet that was first returned in a search since the first synset is usually the most frequently used (Tengi, 1998).

2) POS: the Part-Of-Speech (POS) tagging marked up the linguistic components in sentences of a document for signifying the POS roles of words, which were then used to find the corresponding synset from WordNet.

For example, the word “like” in a document snippet “I really like the history class” was found to have 11 senses in WordNet by using the 1st sense option. The first synset, *{wish.v.01}*, was selected as the matching concept. This synset is a verb synset and its meaning align better with the original meaning of “like” than other synsets.

When used the POS option, POS tagging result resulted in “I/PRP really/RB like/VBP the/DT history/NN class/NN ./.”. The POS tagging of the word “like” is VBP (non-3rd person singular present verb), which can be mapped to the verb sense in WordNet. A search in the WordNet database with string “like” and the sense as a verb returned 5 synsets satisfying these conditions. In the case of more than 1 returned synsets, the first returned synset was selected because that was the most frequently used synset under the particular word + POS conditions.

3.5.1.1.3 Vector Construction Options

The vector construction method was also considered for ONTO-WordNet representation, following the strategies in Hotho et al. (2003a). It addresses what dimensions should be included in the vector space. Three strategies can be applied:

1) concepts only. The vector includes only WordNet synsets as vector dimensions.

2) concepts+words. The vector contains WordNet synsets plus words in the document as vector dimensions.

3) concepts replacing words. A word dimension is replaced by its WordNet synset if the word has a matched synset in WordNet, and a word dimension is kept in the vector if the word has no matched synset.

3.5.1.1.4 Parameters to be Tuned

Within the WordNet synset matching approach there were two parameters that were to be tested: WSD and vector construction. There were 2*3=6 conditions from combining the two parameters. The experimental results will show which combination reaches the best performance, which is later used for comparing with other representation approaches.

3.5.1.1.5 Implementation Details

The author used WordNet database Version 3.0, the most recent version, for synset matching. She employed the JWI package version 2.2.3, a Java library for interfacing with WordNet, to look up synsets of words in WordNet (Finlayson, 2012). For part-of-speech tagging task, she used the OpenNLP packpage (OpenNLP, 2011).

Two more details about the POS strategy of the WSD parameter are illustrated here. First, the POS tag set and the synset senses were not one-to-one matches. There are 36 tags in the Penn Treebank tag set and 5 senses for synset sense. For example, there are different tags for nouns such as NN and NNS in the Penn TreeBank POS tag set, whereas WordNet synset has only one noun sense. The author set up mappings between the POS tags and WordNet sense by starting from the WordNet senses, which have fewer members, and identifying corresponding POS tags in the Penn TreeBank tag set given WordNet senses. Appendix 1 records the mapping between the two sets.

Second, it is possible that a word has multiple synsets for the same POS tag in WordNet, for example, multiple noun senses, and the solution was to use the first

returned synset of that sense in WordNet. As discussed in the “like” example in the WSD option, when the search returns more than one verb synsets subsuming the word “like” (section 3.5.1.1.2), we choose to use the first returned synset.

3.5.1.2 ONTO-Wikipedia

Similar to the WordNet-based representation, concepts in Wikipedia are used to represent documents. First, the author illustrates what is a concept in Wikipedia since it is not as explicitly defined as in WordNet. Second, two concept mapping methods can be applied to matching document text to Wikipedia concepts and are introduced respectively.

3.5.1.2.1 Concepts in Wikipedia

Past research has taken titles of Wikipedia article as concepts and synonyms of Wikipedia titles as concepts (Gabrilovich & Markovitch, 2007; Zhang, 2009). The author endorses Gabrilovich and Markovitch’s (2007) view that a Wikpedia article introduces a concept and the title is thus a concept. As further specified in Gottron, Anderka, and Stein (2011), these concepts are orthogonal and can be used in constructing a Wikipedia concept vector given a document. In Wikipedia database, the “page” table concerns Wikipedia pages, namely the Wikpedia concepts here.

It is noticeable that Wikipedia covers multi-word expressions (phrases) broadly and contains many named entities as well. Though WordNet also contains phrases, it was not addressed in this study because the author used single words in synset matching, and the phrase coverage such as named entities are not as large as Wikipedia. The difference in concept matching methods using WordNet and Wikipedia can also affect the number of concepts identified in a same speech transcript. For

example, the string “human computer interaction” is recognized as a concept by using Wikipedia because it is a Wikipedia article title, whereas it results in three WordNet synsets because the string is split into three single words that are then matched to synsets. Hence the final difference in representation performance may be attributed to the different lengths of concepts of Wikipedia and WordNet.

3.5.1.2.2 Concept Matching Options

Two concept matching techniques were used to match a string of text to Wikipedia concepts. One way was locating concepts from the document text directly; and the other way was indirect, first representing words by a vector of Wikipedia concepts and then representing a document by Wikipedia concepts based on word-concept associations.

1) Direct Matching (DirectWiki). The first way is called direct matching because it identifies Wikipedia concepts directly from text. An intuitive way is to slide a text window of n gram (e.g. 5 gram) in the text to find Wikipedia concepts in the window. This is a simple but error-prone solution because it does not deal with disambiguation issues. For example, the word “Syracuse” has several meanings in Wikipedia, such as concept entries “Syracuse, New York” and “Syracuse, Indiana”. That is to say, if “Syracuse” occurs in a sentence, we need to judge to which meaning it refers, whether the “Syracuse, New York” or the “Syracuse, Indiana”.

The author therefore adopted a Wikipedia disambiguation package, Wikifier, to identify and disambiguate concepts from text (Ratinov and Roth, 2011). Given a text snippet, the package locates strings such as chunks, named entities, and noun phrases by preprocessing and employs a global coherence method to identify the best Wikipedia page match for the located text strings.

2) Explicit Semantic Analysis (ESA). Explicit Semantic Analysis (ESA) is proposed in papers by Gabrilovich and Markovitch (2007; 2009), specifically for representing an arbitrary snippet of text by Wikipedia concepts. ESA makes use of the rich text description of concepts in Wikipedia to build up relations between words and concepts. Given an article page, there is a large text body describing the article title (concept), and therefore words can be associated with concepts from the title and description information. The associations between words and concepts can help establish an inverted index, in which a word is indexed by concepts whose descriptions contain the word. Thus a word can be represented by a vector of concepts associated with it, and a word by concept matrix can be constructed by aggregating concept representation of all words.

3.5.1.2.3 Vector Construction Options

Similar to the vector construction options in ONTO-WordNet, Wikipedia vectors also had these three options: concepts only, concepts + words, concepts replacing words.

1) concepts only. Vector results from DirectWiki and ESA were directly used for this option.

2) concepts + words. For both DirectWiki and ESA, this option means simply merging the Wikipedia vector and word vector.

3) concepts replacing words. The option was only applicable to DirectWiki because DirectWiki can replace words in text by Wikipedia concepts while ESA cannot.

In addition, the author experiments with combining all possible vectors, namely, words, WordNet synsets, and Wikipedia concepts. So there was a 4th option:

4) combine all. Candidate combinations are words+wn1st+DirectWiki and words+wnpos+DirectWiki.

3.5.1.2.4 Parameters to be Tuned

For concept matching in ONTO-Wikipedia experiments, the author tried two options: DirectWiki and ESA. Since DirectWiki employed the Illinois Wikifier package (Ratinov and Roth, 2011) with its default setup, it did not involve parameter setup. For the ESA method, the author needed to set up a parameter *n* that limits the number of Wikipedia concepts to be used in the vector representation. She chose to test with n=10, 20, 50, 100, 1000 respectively.

3.5.1.2.5 Implementation Details

For ESA method, the version of Wikipedia in use was the 2010 September dump, freely available online. The dump was stored in a local MySQL database, with 3,563,430 Wikipedia concepts in total. Concepts were pruned as described in Gabrilovich's (2007) dissertation thesis, which pruned unimportant concepts by a series of steps such as removing concepts with less than 5 incoming and outgoing links, dropping concepts whose text had less than 100 non-stop words, and removing concepts which had weak associations with a word based on tfidf values. After pruning, 2,725,469 concepts remained and they were used for ESA vectors. Gabrilovich's (2007) and Zhang's (2009)'s experiments resolve redirect links and makes redirect concepts as a same concept, however this study did not resolve redirects.

3.5.2 Ontology-based Representation and Reasoning Approach (OntoReason)

This approach shares the same process as the ONTO approach on representing documents using ontology concepts, and in addition, it also deals with the unknown term issue. Following the ONTO approach, it first represented text in vector of ontology

concepts, and then unknown concepts were identified from comparing test and training vectors. If a concept present in a test vector did not occur in the training vector (derived from training corpus), then this concept was called an “unknown concept”. This unknown concept was ignored when computing similarity between test and training vectors. This may have the potential risk of underrepresenting important concepts in test documents and further on causes inaccurate scoring. OntoReason aims to resolve the problem by estimating the importance of the unknown concept in training corpus by using ontological knowledge.

As a solution, a dimension of that unknown concept was added to the training vector. It was completed by computing the semantic similarity between known concepts in training vector and the unknown concept and then reasoning the weight of the unknown concept by using concept similarity information. As a knowledge source, ontology can be used to compute semantic similarity between concepts. Then the weight of the unknown concept was derived and the dimension of the unknown concept was added to the training vector. The semantic similarity computation varies according to different ontologies, and the WordNet and Wikipedia based reasoning approaches are introduced respectively below. Note that this method is designed specifically for generating input vector for the e-rater machine learning model.

Reasoning Strategy

The author uses a concrete example to illustrate the reasoning process (in Figure 7). This part is applicable to both WordNet and Wikipedia ontologies while the similarity calculation part needs to be customized. The score level vector, a vector generated from concatenated transcripts of a same score level, does not contain Concept C_3 from

the test vector, and therefore C_3 is an unknown concept to the score level. We need to figure out the weight of C_3 in the score level and test vectors to add C_3 to the vectors.

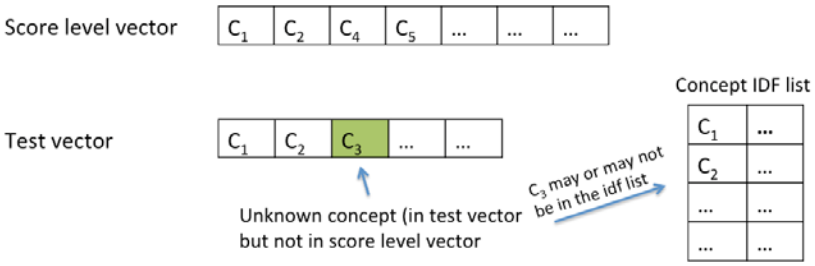


Figure 7. Unknown concept example.

For the score level vector, the weight of C_3 is estimated by using weight information of other concepts in the vector. For the test vector, we face two possibilities: 1) C_3 is present in the idf list, meaning the concept occurs in other score levels though not in this score level. The weight of C_3 is simply multiplying its tf in the test transcript by its idf vaue. 2) C_3 is absent from the idf list, which means it is not in any score levels. Since its tf is known, we only need to estimate its idf, which is computed by using idf information of other concepts in the test vector.

3.5.2.1 OntoReason-WordNet

3.5.2.1.1 Identifying Unknown Concepts

The documents were first converted to vectors of WordNet synsets as described in the ONTO-WordNet in the ONTO approach. The representation also followed the best parameter options resulted from the ONTO-WordNet experiment. Then for a test vector that was to be compared with the 3 score level vectors, we found unknown concepts for each score level. Henceforth unknown concepts varied when a test vector was compared with different score level vectors.

3.5.2.1.2 Concept Matching Options

Similar to ONTO-WordNet, there were two options of identifying synsets from text and they were experimented in OntoReason-WordNet too:

1) wn1st. Use the 1st returned synset as matched synset.

2) wnpos. Find matched synset based on word string and its POS role in the original sentence.

3.5.2.1.3 Concept Similarity Options (plus reasoning details)

The first two options, Path similarity and Lin similarity, make use of similarities between known and unknown concepts to guess weights of unknown concepts. Taking advantage of the hierarchical structure of WordNet, the similarity computation can be edge based (Path similarity) or information content based (Lin similarity). The last one, default similarity, adopts a simple way of averaging weights of all concepts to calculate the weight of the unknown concept.

1) Path similarity. It measures the length of the path from one concept to another concept in WordNet. It is the inverse of the shortest path between the two concepts (Pedersen et al., 2004). It is used for computing similarity between two WordNet synsets.

The weight of the unknown concept in the score level vector is calculated by averaging weights of its n similar concepts in the score level vector:

$W_{Unknown} = (\sum_i W_{C_i})/n$ (for unknown concept in score level vector)

where $i \in (1, \ldots, n)$, C_i are top n concepts in the score level vector with the largest similarity to the unknown concepts, and n is set to 5 in this study. In other words, the method first computes similarity between the unknown concept and each concept in the score level vector, ranks the similarity and finds the top 5 concepts with the largest

similarity, and makes the average weight of these 5 concepts as the estimated weight of the unknown concept.

If the unknown concept is not in the idf list, then its idf value is estimated by:

$idf_{Unknown} = (\sum_i idf_{C_i})/n$ (for unknown concept's idf in test vector)

where $i \in (1, \dots, n)$, C_i are top n concepts in the idf list with the largest similarity to the unknown concepts, idf_{C_i} is the idf value of C_i, and n is set to 5 too.

In this way, the score level and test vectors are expanded by adding unknown concepts to them. Following this, the e-rater scoring model can be applied to assign scores based on the expanded representation.

2) Lin's (1998) similarity. It is based on taxonomy structure like WordNet and word probability in external corpus, by computing semantic similarity between two concepts of a taxonomy. It is also used for computing WordNet synsets here, and the formula is:

$$sim(C_1, C_2) = \frac{2 \times log\, P(C_0)}{log P(C_1) + log P(C_2)}$$

where C_1 and C_2 are two arbitrary concepts, C_0 is the most specific concept subsuming C_1 and C_2 in WordNet, and P(C) is the probability that a randomly selected object belongs to concept C. P(C) can be derived by processing an external corpus and counting its relative frequency in the corpus (Lin, 1998).

The unknown concept's weight in the score level vector and its idf in test vector are calculated in the same way as in Path similarity:

$W_{Unknown} = (\sum_i w_{C_i})/n$ (for unknown concept in score level vector)

where $i \in (1, \ldots, n)$, C_i are top n concepts in the score level vector with the largest similarity to the unknown concepts, w_{C_i} is the weight of C_i , and n was set to 5 in this study.

If the unknown concept is not in the idf list, then its idf value is estimated by:

$idf_{Unknown} = (\sum_i idf_{C_i})/n$ (for unknown concept's idf in test vector)

where $i \in (1, \ldots, n)$, C_i are the top n concepts in the idf list with the largest similarity to the unknown concepts, idf_{C_i} is the idf value of C_i, and n is set to 5.

3) Default similarity. This option does not compute similarity between known and unknown concepts, but just uses average weight of the known concepts as the estimated weight of the unknown concept. It is designed to test the usefulness of similarity computation compared to a simple average method. Thus the weight of unknown concept is:

$W_{Unknown} = (\sum_i w_{C_i})/m$ (for unknown concept in score level vector)

where C_i are all the concepts in the score level vector, w_{C_i} is the weight of C_i , and m is number of concepts in the score level.

If the unknown concept is not in the idf list, then its idf value is estimated by:

$idf_{Unknown} = (\sum_i idf_{C_i})/m$ (for unknown concept's idf in test vector)

where C_i are all the concepts in the idf list, idf_{C_i} is the idf value of C_i, and m is number of concepts in the score level.

For example, a sentence from a test file is "so radio also create a great impact on this uh people communication". The words are matched to WordNet synsets, and the concept {*impact.n.01}* is found to be an unknown concept to the score level vector. Among the concept dimensions of the training vector, under the Path similarity option,

the five most similar concepts to the unknown concept are {*happening.n.01}*, {*event.n.01}*, {*change.n.01}*, and two others. The weight of the unknown concept {*impact.n.01}* in the score level vector is the average weight of the five similar concepts.

3.5.2.1.4 Implementation Details

The author adopted a Java package called Java WordNet::Similarity for computing Path and Lin similarities in WordNet (Hope, 2008). It is a package specifically for calculating different types of semantic similarity in WordNet, and is a Java version of the WordNet::Similarity Perl module (Pedersen, Patwardhan, Banerjee, & Michelizz, 2008).

3.5.2.1.5 Parameters to be Tuned

The author tested with concept matching and concept similarity parameters. The best single combination was chosen and used as the basis for between-approach comparison.

3.5.2.2 OntoReason-Wikipedia

As discussed above, OntoReason-Wikipedia shared similar workflow as OntoReason-WordNet while they differed in their similarity computing details.

3.5.2.2.1 Identifying Unknown Concepts

Firstly the documents were represented in vectors of Wikipedia concepts as discussed in the ONTO-Wikipedia representation in the ONTO approach. Only the DirectWiki option was employed for representation and reasoning, and ESA was not applicable for the reasoning case because it does not directly extract concepts from text. After obtaining the vectors of Wikipedia concepts, the author identified the unknown concepts in the same way as in the OntoReason-WordNet approach.

3.5.2.2.2 Concept Similarity Options (plus reasoning details)

1) Content based similarity.

Unlike WordNet, Wikipedia is loosely structured and therefore the path based and information content based similarity does not apply to the Wikipedia case. Here the text descriptions of Wikipedia concepts were utilized to compute concept similarity. Given two Wikipedia concepts, their text descriptions are represented by vectors of words, and then their similarity is the cosine similarity of the two word vectors.

For example, given two Wikipedia concepts, C_1 and C_2, their text descriptions are converted to two vectors of words as $V_1=(w_{11}, w_{12}, \ldots, w_{1n})$, $V_2=(w_{21},w_{22}, \ldots, w_{2n})$, and the similarity between C_1 and C_2 is calculated as:

$$sim(C_1, C_2) = CosSim(V_1, V_2) = \frac{\sum_{i=1}^{n} w_{1i} w_{2i}}{\sqrt{\sum_{i=1}^{n} {w_{1i}}^2} \sqrt{\sum_{i=1}^{n} {w_{2i}}^2}}$$

This similarity method helps guess weights of unknown concepts, in a similar way to OntoReason-WordNet. First, weight of the unknown concept in a score level vector is computed by:

$w_{Unknown} = (\sum_i w_{C_i})/n$ (for an unknown concept in a score level vector)

where C_i are the top n concepts in the score level vector with the largest similarity to the unknown concepts, w_{C_i} is weight of C_i in the score level vector, and n is set to 5 in this study.

Second, if the unknown concept is not in the idf list, then its idf is calculated as:

$idf_{Unknown} = (\sum_i idf_{C_i})/m$ (for unknown concept's idf in test vector)

where C_i are the top n concepts in the idf list with the largest similarity to the unknown concepts, idf_{C_i} is the idf value of C_i, and n is set to 5 in this study.

2) Default similarity.

The logic is also similar to the default similarity option in OntoReason-WordNet. Instead of finding similar concepts to the unknown concepts, we simply average weights of known concepts to derive the weight of an unknown concept. Specifically, the weight of an unknown concept is:

$w_{Unknown} = (\sum_i w_{C_i})/m$ (for unknown concept in score level vector)

where C_i are all the concepts in the score level vector, w_{C_i} is the weight of C_i in the score level vector, and m is number of concepts in the score level.

If the unknown concept is not in idf list, then its idf value is estimated by:

$idf_{Unknown} = (\sum_i idf_{C_i})/m$ (for unknown concept's idf in test vector)

where C_i are all the concepts in the idf list, idf_{C_i} is the idf value of C_i, and m is number of concepts in the score level.

3.5.2.2.3 Parameters to be Tuned

One parameter, concept similarity method, was to be tested. A best option was to be selected based on scoring model performance under different reasoning methods.

3.5.2.2.4 Implementation Details

Text description of Wikipedia concepts contains HTML tags and MediaWiki markups, which need to be cleaned before computing Wikipedia concept similarity. The author used the WikipediaExtractor, a python script for cleaning tags and markups in Wikipedia page (Attardi & Fuschetto, 2013).

In converting text description to a vector of words, the weight of a word was its raw frequency (tf) in the Wikipedia text for effectiveness of computing. The content-based similarity between concepts was then calculated from the tf vectors.

3.6 Building Scoring Models from the Representations

Machine learning methods were applied on the speech transcript representations to build scoring models and predict scores. Machine learning technique is suitable for scoring models because the representation approaches all result in vectors, which can be directly used as input features for machine learning models. The author employed e-rater model as the primary machine learning model and also experimented with Naïve Bayes model in some cases. According to the training-testing paradigm, training documents were used to generate a scoring model and test documents were used to evaluate the predictiveness of the model.

3.6.1 E-rater Model

The e-rater model tackles two content features for automatic essay scoring, which are *max.cos* and *cos.w4*[3]. The computation of max.cos is similar to a machine learning processing that assigns a class label to an instance, and thus the author takes the max.cos value as machine learning results and the process of calculating max.cos as a machine learning model, called “e-rater model”. Therefore the max.cos calculation process is treated as a machine learning model, and then the values of max.cos and cos.w4 are used for correlation analysis for evaluation purpose. The author would like to distinguish between *max.cos value* and *max.cos correlation* here, with the former meaning a computation based on content vector to predict speech scores and the latter referring to the correlation between max.cos values (predicted scores) and actual scores. On the other hand, because cos.w4 values are solely used for correlation analysis, it is introduced later in section 3.7.4.

[3] As mentioned in footnote [1] in section 3.1, max.cos and cos.w4 are “content features”, which is a typical naming convention in the automated scoring literature, reflecting a variable in a construct (e.g. speaking proficiency). “Feature” in machine learning has a different meaning, which is a vector dimension rather than a variable.

Salton et al.'s (1975) content vector analysis is used in e-rater to compute the max.cos content feature in e-rater. The assumption is that "good essays will resemble each other in their word choice, as will poor essays" (Attali & Burstein, 2006). In the vector space, the closer an essay is to another essay then a similar score should be assigned. The distance between two essays is measured by the cosine similarity of the vectors.

The *max.cos* feature compares a test document's similarity to a score level group to decide content wise how similar the document is to the documents of a specific score level. In the data set of this study, there are 3 score levels, a test document is compared to the training documents of each score level (score level 2, 3, 4) to find which score level training documents it is most similar to.

More specifically, the training documents belonging to a same score level are used to generate score level vectors by aggregating the training documents (resulting in score 2 vector, score 3 vector, score 4 vector[4] in this study). The score level and test vectors can be derived from any of the representation approaches in sections 3.4 and 3.5. Given a test document, after it is converted to vector representation, its similarity to the score level vectors are ranked and the score level vector that has the largest similarity to the test vector is selected as the value of the max.cos feature. Moreover, the selected score value is assigned to the test document as the predicted score of the document. For example, if a test vector is most similar to the score level 3 vector in vector space, then the test document is scored 3 because of their proximity in space.

[4] In subsequent sections, the author continues to use this naming convention, "score level i vector", to refer to the vector obtained from aggregating all the transcripts of score level i, or the "super vector" mentioned in Burstein (2003).

E-rater uses cosine similarity to compute similarity between test vector and score level vectors. Given the vector representation of a test document $(w_1, w_2, \dots, w_n)$ and the vector of score level s training documents $(w_{s1}, w_{s2}, \dots, w_{sn})$ (s=2,3,4), their similarity is computed as:

$$\frac{\sum_{i=1}^{n} w_i w_{si}}{\sqrt{\sum_{i=1}^{n} {w_i}^2}\sqrt{\sum_{i=1}^{n} {w_{si}}^2}}$$

The e-rater machine learning model can be formalized as:

$$pscore = \underset{s}{argmax} \frac{\sum_{i=1}^{n} w_i w_{si}}{\sqrt{\sum_{i=1}^{n} {w_i}^2}\sqrt{\sum_{i=1}^{n} {w_{si}}^2}} \ (s = 2,3,4)$$

where pscore is the predicted score of the test document and other symbols retain the same meaning as above.

The max.cos value is an integer since it is a score level, and this value is also the machine learning result, which performs e-rater (max.cos) calculation and assigns this score level to the test transcript as its predicted score.

The mechanism of e-rater is similar to the Rocchio classifier described in Joachims (1997), which is a text classifier based on Rocchio relevance feedback (Rocchio, 1971). Rocchio classifier aggregates document vectors belonging to the same class to derive a prototype vector for each class, and test documents are classified according to their distances to these prototype vectors. As we can see, the e-rater model is similar to the mechanism of Rocchio classifier.

3.6.2 Naïve Bayes (NB) Model

Naïve Bayes (NB) is a frequently used model in machine learning and text categorization. In the context of text mining, it is a probabilistic model estimating probability of a document belonging to a class. One important assumption of NB is the

independency between terms, meaning the probability of one term is not conditional on another term.

NB model calculates the probability of a document d belonging to a class c in this way (Manning, Raghavan, Schutze, 2008):

$$p(c|d) = \frac{p(c) * \prod_{i=1}^{n} p(w_i|c)}{p(d)}$$

where p(c|d) is the probability of belonging to class c given document d, p(c) is the prior probability of class c, n is the number of tokens of document d, w_i is a token in the vocabulary of document d, $p(w_i|c)$ is the probability of token w_i occurring in class c, and p(d) is the probability of document d. Since p(d) is a constant given a particular document, it is crossed out from the equation for convenience of calculating. Thus the probability can be simplified as:

$$p(c|d) = p(c) * \prod_{i=1}^{n} p(w_i|c)$$

The NB model computes the probability of a document belonging to each class and selects the class with the largest probability as the class of the document. The process of assigning a class to a document can be formalized as:

$$\underset{c}{argmax}\, p(c) * \prod_{i=1}^{n} p(w_i|c) \quad \text{(Equation 1)}$$

where the symbols shares the same meaning as the two equations above.

In the context of this study, the class of a document is the score level that it belongs to and the task of NB model is to predict the score level of a test document. First, the NB scoring model is built using the training documents by figuring out model components including p(c) and $p(w_i|c)$. After representing documents in vector style, $p(w_i|c)$ can be computed from document vectors whose class is c. The values of the

document vector depend on the representation approach and weighting scheme. For example, it can be Boolean weighting in the BOW representation, along with other weighting and representation combinations. Then given a test document its class can be predicted based upon Equation 1.

3.7 Evaluating Scoring Models and Representation Approaches

3.7.1 3-fold cross-validation

The data set was partitioned into 3 folds using stratified splitting, and therefore there were 3 rounds of model building and model testing which result from the 3 folds. The machine learning model is eventually evaluated using results of model performance of each round. The predicted scores were recorded after each round and results from the 3 rounds together formed a confusion matrix that was further used to compute F measure and accuracy, following Weka software's practice (Hall et al., 2009). The aggregated results can also be used to calculate other measures including correlation and kappa.

Figure 8 takes confusion matrix as example and shows how results from different runs were aggregated to the final confusion matrix, which contains information of model prediction results compared to the actual classes. In the 1st round of machine learning, data folds 1 and 2 were used as training set from which a machine learning model is generated and fold 3 is test set, and the resulting confusion matrix was confusion matrix a in Figure 8. Similarly, confusion matrices b and c were obtained from 2nd and 3rd rounds. The cells of the final confusion matrix were derived by summing the corresponding cells of confusion matrices a, b, and c, as shown in Figure 8.

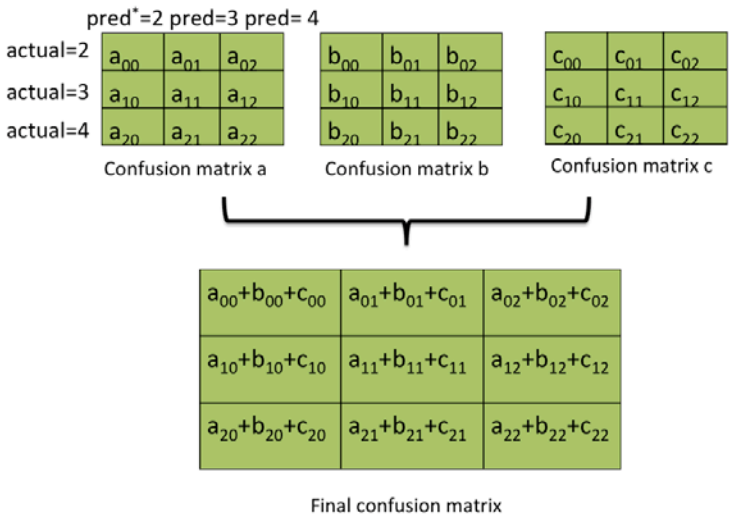


Figure 8. Aggregating confusion matrix from each run to form the final confusion matrix. (pred*=predicted)

The correlation and kappa evaluation for cross validation was conducted in a similar way. It first aggregated prediction results from the test set of each round so that each document in the original data set had a predicted class, and then Pearson correlation or kappa was computed between predicted and actual scores using these two scores of all the documents.

3.7.2 Evaluating Scoring Models

The author chose 5 evaluation measures for measuring results from different aspects: 2 are evaluation for general classification models (F measure and accuracy), 2 are evaluation for ordinal classification (max.cos correlation and kappa), and 2 is for evaluating a content feature that is produced from representation approaches (max.cos correlation and cos.w4 correlation) using correlation analysis.

F Measure

F measure is a prevalent way of evaluating machine learning models and is used here to indicate their predictivenss on speaking scores. This is a multi-class machine

learning problem and the author chose to use macro-averaged F measure for evaluation, based on the formula provided by Özgür, Özgür , and Güngör (2005) as shown below

$$F = \frac{\sum_{i=1}^{n} F_i}{n}$$

where F_i is the F measure of class I, and n is the number of classes.

A popular machine learning toolkit, Weka (version 3.6), calculates weighted macro-averaged F measure, which adds a weight for each F_i according to number of instances of this class (Hall et al., 2009). But since each class was treated equally here, the author computed the average instead of the weighted average to reflect the model's average performance on each class.

Moreover, the F measure of class i was computed as (Croft et al., 2010; Özgür et al., 2005):

$$F_i = \frac{2R_i P_i}{(R_i + P_i)}$$

where R_i is recall of class i (percentage of instances correctly predicted as class i out of all true instances in class i) and P_i is precision of class i (percentage of instances correctly predicted as class i out of all instances predicted as class i), which are respectively obtained by:

$$R_i = \frac{TP_i}{(TP_i + FN_i)} \quad and \quad P_i = \frac{TP_i}{(TP_i + FP_i)}$$

where TP_i is true positive for class i, FN_i is false negative for class i, and FP_i is false positive for class i (Özgür et al., 2005).

Accuracy

This is also an evaluation metric from machine learning, without considering ordinal class information. Out of all the classified instances it calculates how many are correctly classified. Given a confusion matrix M, with row as actual values and columns standing for predicted values, accuracy is computed as

$$accuracy = \frac{\sum_{i=1}^{n} M_{ii}}{\sum_{i=1}^{n} \sum_{j=1}^{n} M_{ij}}$$

where n is the total number of classes.

Correlation Analysis

Scoring models can also be evaluated by correlation analysis because features (in the sense of automatic scoring) are often evaluated by their correlations with human assigned scores (Cucchiarini et al., 2002; Dodigovic, 2009; Zechner et al., 2009). A higher correlation with human scoring indicates better predictiveness of that feature on speaking proficiency. Similarly, we run correlation analysis between the two content features in e-rater (as illustrated below) and actual scores. For clarification, max.cos and cos.w4 are the two content features computed from the content vectors, and their correlations with the actual scores, namely max.cos correlations and cos.w4 correlations, are used as an evaluation indicator for scoring performance.

The author used Pearson's r for correlation analysis, because it is a typical evaluation method in the automatic scoring field, as used in these studies (Cucchiarini et al., 2002; Dodigovic, 2009; Zechner et al., 2009). A higher correlation indicates better model performance, and thus we prefer a representation approach resulting in higher correlation.

max.cos correlation. The max.cos content feature measures to which score level group the test file is most similar in vector space by comparing cosine similarity (Attali & Burstein, 2006). The computation max.cos value has been described earlier in section 3.6.1, and the output max.cos values are integers.

Max.cos value is a content feature from the automatic scoring perspective as well as the predicted score of a test transcript from the machine learning perspective. Max.cos correlation computes the Pearson correlation between max.cos values (predicted score) and actual scores. Besides being an evaluation metrics for content feature max.cos, max.cos correlation is also an evaluation for ordinal classification from machine learning perspective .

cos.w4 correlation. Cos.w4 content feature measures how close a test document is to the highest score level group (Attali & Burstein, 2006). For instance, the highest score is 4 in the data set, and cos.w4 value is derived by computing cosine similarity between a test vector and score level 4 vector. Since it is similarity, cos.w4 values are real numbers. Then given a test set, cos.w4 correlation is derived by calculating Pearson correlation between cos.w4 values and the corresponding actual scores.

An example. Figure 9 illustrates how max.cos and cos.w4 values are computed. Given a test vector generated from a test transcript, along with three score level vectors generated from the training set, the cosine similarity between the test vector and each score level vector is computed. It results in 3 similarity values: 0.2 with score 2 vector, 0.8 with score 3 vector, and 0.5 with score 4 vector. The max.cos value is the score level with the largest similarity to the test vector, which is score level 3 in this case,

since its similarity 0.8 is the highest value. The cos.w4 value is 0.5 here, since it is the similarity between the test vector and score level 4 vector.

Note that there are further steps for computing correlations, since what we obtained above is the values of max.cos and cos.w4, rather than the max.cos and cos.w4 correlations. For correlation calculation, for example for max.cos correlation, given a test set, we aggregate max.cos values of all the test transcripts and their corresponding actual scores, and run Pearson correlation between the max.cos values and actual scores to derive max.cos correlation. The cos.w4 correlation can be derived in a similar way.

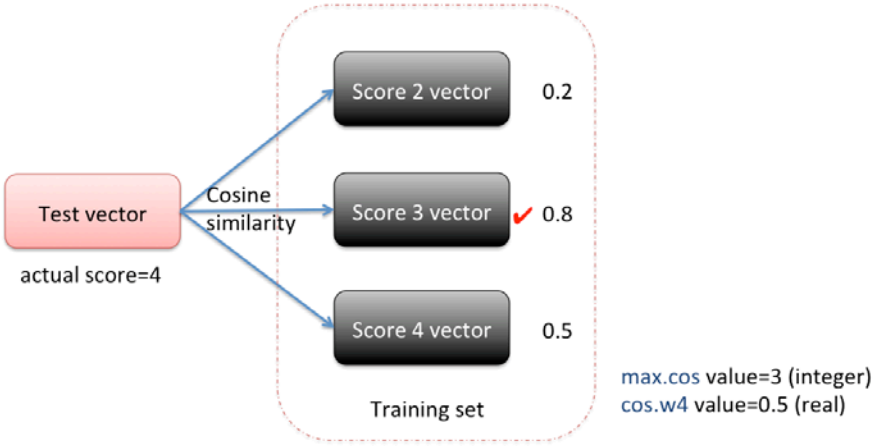


Figure 9. Computing max.cos and cos.w4 values, the pre-step of computing max.cos and cos.w4 correlations.

Kappa Analysis (quadratic weighted kappa)

Kappa measures inter-rater agreement between two raters (Banerjee et al., 1999), which in this context, is the agreement between predicted scores and actual scores of speech transcripts. It basically measures to what degree automatic scoring agrees with human scoring above chance-level. Since the score levels are ordinal, weighted kappa is suitable for measuring inter-rater agreement because it considers the difference between disagreements. For example, given a transcript with actual score of

4, predicting it to score 2 and score 3 are different because score 3 is closer to score 4 than score 2. Quadratic weighted kappa measures the disagreement between predicted and actual scores by assigning quadratic weights to disagreements.

3.7.2 Evaluating Effects of Representation Approaches

The evaluation of representation approaches relies on the evaluation of scoring models. As shown in Figure 10, documents can be represented in four different ways in this study and the representation outputs are sent to machine learning models for building scoring models. The scoring models are evaluated by the 5 measurements introduced in section 3.7.1. When using the same machine learning method, the differences between representation approaches attribute to the differences in scoring model performance. The effectiveness of the 4 representation approaches is comparable when the machine learning method is fixed. The major machine learning model is e-rater, and Naïve Bayes is only used in some cases to see whether it enhances performance over e-rater model, more of a comparison between machine learning models.

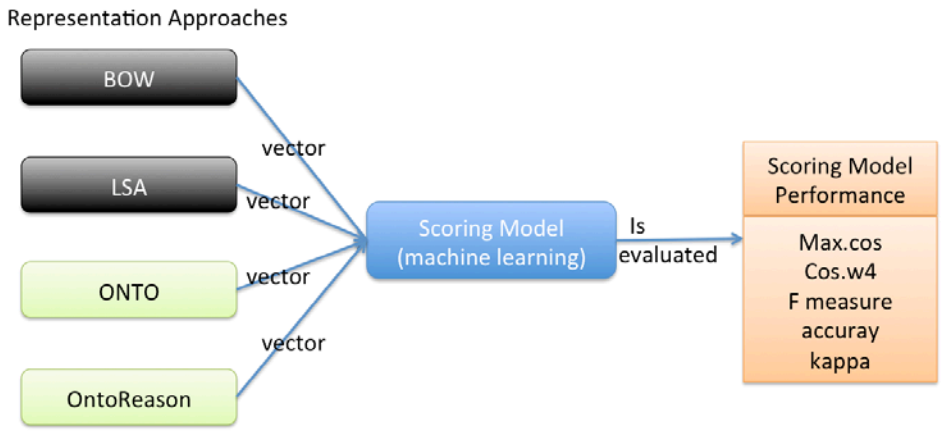


Figure 10. Evaluating representation approaches.

3.8 Summary

The majority of the chapter presents the two baseline systems and two experimental systems with representation details and parameter options, and meanwhile it delineates the scoring models and evaluation methods (summarized in Table 5). Experiment results from the systems are used to decide best parameter options, compare performance of scoring models, and further compare between representation approaches. Chapter 4 presents the results from preliminary implementation and relevant analysis.

Table 5. Summary of Baseline and Experimental Systems

System	**Represent-ation**	**Ontology Specific Representation**	**Scoring Models**	**Parameters to be Tested**	**Evaluation**
BOW	BOW	n/a	e-rater, NB	weighting	All 5 measures
LSA	LSA	n/a	e-rater	Rank k for matrix approximation	All 5 measures
ONTO	ONTO	ONTO-WordNet	e-rater, NB	WSD strategy, vector construction strategy	All 5 measures
		ONTO-Wikipedia	e-rater	Concept matching method	All 5 measures
OntoReason	OntoReason	OntoReason-WordNet	e-rater	Concept similarity	All 5 measures
		OntoReason-Wikipedia	e-rater	Concept similarity	All 5 measures

4. ANALYSIS

4.1 Overview

Results from the experiment were analyzed from two perspectives: parameter analysis that compared performance of different parameters on the same representation approach and hypothesis analysis that compared performance between different representation approaches. The analysis also provides basis for examining the hypotheses mentioned in section 3.3. In other words, the parameter analysis was conducted to evaluate the within-approach performance whereas the hypothesis analysis was for between-approach comparisons. Parameter analysis not only provides understanding of effects of parameters, but also of the mechanism of each approach, which further offers insights into different approaches for between-approach comparisons.

Both types of analyses compare performance results through a number of measurements, including max.cos correlation, cos.w4 correlation, F measure, accuracy rate, and kappa (see details in section 3.7). The benefit of using multiple measurements is that they check performance from various perspectives to allow for a comprehensive evaluation. Each measurement method has its own focus and purpose and evaluates a representation approach from a particular aspect.

As Figure 11 shows, kappa and max.cos correlation are measurements tailored for ordinal classification where class labels are ordinal. The F measure and accuracy rate measure the performance of machine learning, assuming that class labels are nominal. Max.cos correlation and cos.w4 correlation measure performance from the perspective of correlation by computing correlations between a content feature

(max.cos or cos.w4) and human scoring, which are the typical way of evaluating features in the automatic scoring field and an indirect evaluation of representations in this study. The max.cos correlation, F measure, accuracy rate, and kappa measurements all deal with to what extent the predicted scores are different from actual scores. The smaller the difference is between the predicted and the actual scores, the better the representation approach performance will be. These measurements are all based on the predicted scores from machine learning models (e-rater or Naïve Bayes). Unlike the other four measurements, the cos.w4 correlation does not need predicted scores but rather, only needs cos.w4 content features and actual scores for the computation. This means that cos.w4 correlation does not rely on the output of machine learning model to perform evaluation computation. Together, these five measurements present a holistic evaluation picture for representation approaches.

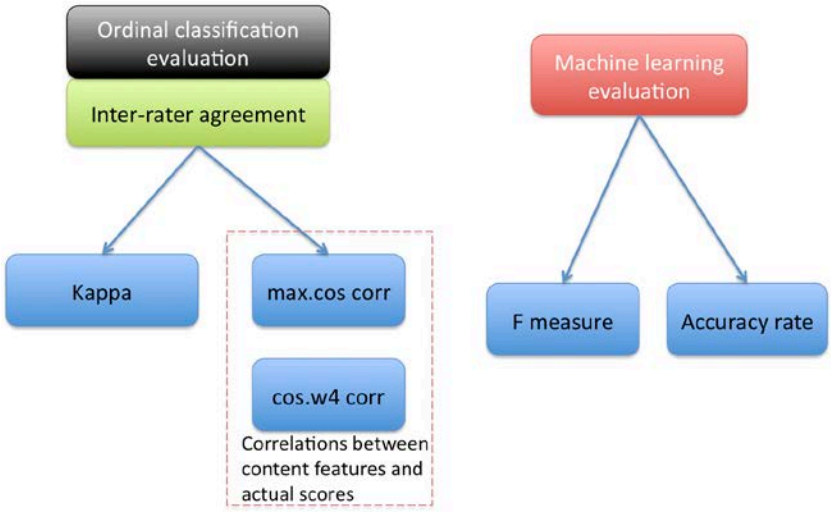


Figure 11. Evaluation measures and their evaluating perspectives.

The author then conducted in-depth analysis such as analysis of vectors and selected case. From the data, exemplar transcripts were selected for vector and case analyses in order to summarize patterns and interpret results. Parameter and

hypothesis analyses inspect results from a macro perspective by comparing performance at the prompt level, whereas vector and case analyses of in-depth analysis examine, at individual transcript level from a micro perspective: content, vectors, and term weights of some transcripts. A significance t-test and prompt-specific analysis were also conducted on the performance results.

4.2 Parameter Analysis (within-approach analysis)

The within-approach analysis deployed evaluation measures on each prompt, the values from which were averaged to obtain an average performance among the four prompts. The average performance of a representation approach was the focus of this analysis. The within-approach analysis describes the overall performance of an approach and balances the randomness for individual prompts. Given the large number of individual prompt evaluation results, the inspection of representation performance on individual prompts was focused primarily on reporting and analyzing abnormal patterns.

4.2.1 Bag-of-Words (BOW) Parameters

The weighting method parameter was evaluated for the BOW representation approach with two weighting options: normalized tfidf and normalized tf. They were each coded as BOW(tfidf) and BOW(tf), respectively. Term frequencies in documents were first obtained for both options, then multiplied with BOW(tfidf) by their corresponding idf values (BOW(tf) needed not multiply by idf). Vectors generated from both options were normalized through dividing by the Euclidean length to obtain a vector of length 1. Results of the two options are displayed in Table 6 below.

Table 6. BOW results.

	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
BOW(tfidf)	**0.3494**	**0.3556**	**0.4627**	**0.4786**	**0.3441**
BOW(tf)	0.2259	0.0806	0.3804	0.3789	0.2178

These results show that BOW(tfidf) performed better than BOW(tf) did on all measurements. One possible reason for the lower performance of BOW(tf) was that, in BOW(tf), many words shared the same weights. Since the speech transcripts were relatively short in length (121.19 words on average), a word usually occurred only once or twice, which led to many words bearing the same term frequencies. Words of the same frequency shared the same weight even after normalization due to the fact that idf was not used to adjust weights. This made it difficult to distinguish between important and unimportant words, possibly causing lower performance of BOW(tf).

In addition, having inspected the confusion matrices that show actual and predicted score levels of transcript instances, it was found that the scoring models from BOW tended to classify instances as score 3. This happened to instances of prompts 098 and 099 in which, regardless of the actual score level, more instances were classified to score 3 than to score 2 and 4. We would hope at each score level, the majority of the instances were classified to that score level for a better performance (higher recall rate).

For example, in prompt 098 under BOW(tfidf): out of the 83 instances for actual scores of 2 (the 2nd row of Table 7), 45 were classified to score 3 and the remaining 38 to scores 2 and 4; 85 out of the 157 score=3 instances were predicted as score=3, 26 as score=2 and remaining 46 as score=4. Out of the 78 score=4 instances, 45 were

classified to score 3, only 1 as score 2, and the other 32 as score 4. These results present an unbalanced classification.

Table 7. Confusion matrix for prompt 098, using BOW(tfidf).

	Score=2 (predicted)	Score=3 (predicted)	Score=4 (predicted)	Sum
Score=2 (actual)	30	45	8	83
Score=3 (actual)	26	85	46	157
Score=4 (actual)	1	42	35	78
Sum	57	172	89	318

Best Parameter Option (tfidf)

Because BOW(tfidf) outperformed BOW(tf) on each measurement, BOW(tfidf) was chosen as the best parameter option and used to represent BOW approach performance in hypothesis analysis.

4.2.2 Latent Semantic Analysis (LSA) Parameters

The LSA parameter, k, defines matrix dimensions cutoff thresholds as the number of concept dimensions used for LSA representation. The experiments were run with k=10, 20, 30, 40, 50, 100, and 200 respectively. Table 8 lists averaged performance measures for each k option; these results are also charted in Figure 12 to visually show the trends when parameter *k* changes. From the results it was observed that:

1) All measures, except the cos.w4 correlation, exhibited a similar trend that the LSA parameter values increased from k=10 to around k=40 or 50 and decreased after they reached the peak k=40 or 50. The peak values for the 5 measures happened in k=40 (max.cos correlation, accuracy, kappa) and k=50 (F measure). The parameter values fluctuated without any pattern as k value increased.

2) On the contrary to finding 1, cos.w4 correlation showed an overall increasing trend, which first decreased at k=20 and then gradually increased with some fluctuation.

3) Overall LSA approaches had low cos.w4 correlation performance, especially when compared to max.cos correlation that is a similar measurement. The lowest cos.w4 correlation value was 0.0823 at k=20, and the highest value was 0.1791 at k=200, which was a fairly low correlation compared to max.cos correlation, of which the lowest value was 0.1751 at k=10.

4) The low values of cos.w4 correlation may indicate that representing transcripts using LSA vectors disrupted cos.w4 correlation measurement. Since cos.w4 correlation essentially measured the associations between distance from a test transcript to the best quality transcripts and its actual score. Theoretically a better representation should: draw test transcripts that are actually scored as 4 closer to the score level 4 vector, draw transcripts with actual score of 2 or 3 farther from the score level 4 vector, and, therefore, better representations should result in a higher cos.w4 correlation. However, LSA results turned out low cos.w4 correlations that, moreover, were lower than BOW(tfidf). It seems LSA representation did not realize the goal of keeping good transcripts closer to best quality sample transcripts.

Table 8. LSA performance.

	Avg. max.cos corr.	**Avg. cos.w4 corr.**	**Avg. F measure**	**Avg. accuracy**	**Avg. kappa**
LSA (k=10)	0.1751	0.1618	0.3533	0.4116	0.1496
LSA (k=20)	0.2242	0.0823	0.3848	0.4204	0.2039
LSA (k=30)	0.2003	0.1016	0.3727	0.3852	0.1927
LSA (k=40)	**0.2506**	0.151	0.3931	**0.4442**	**0.2393**
LSA (k=50)	0.228	0.1053	**0.412**	0.445	0.2225
LSA (k=100)	0.2394	0.1395	0.368	0.3898	0.2272
LSA (k=200)	0.1998	**0.1791**	0.3664	0.392	0.1887

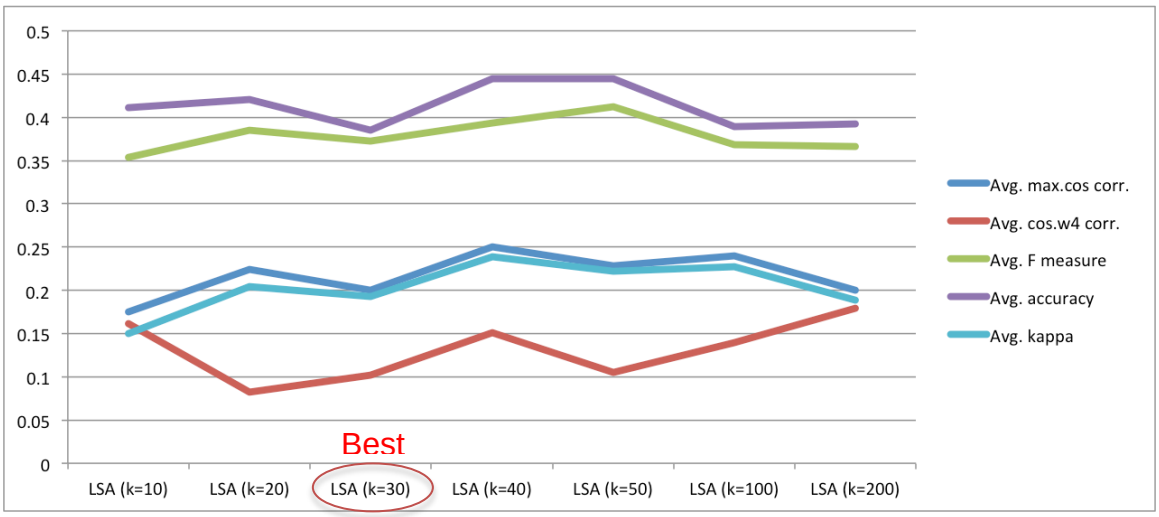


Figure 12. LSA performance from different k options.

Best Parameter Option (k=40)

As discussed in point 3, when k=40, three of the five measures reached their highest value, and thus the author selected this option as the best parameter.

4.2.3 ONTO-WordNet Parameters

Two parameters, WSD strategy and vector construction strategy for concept matching, were tuned in this experiment. Tables 9 and 10 below list possible options for these parameters and experimental results for each parameter setup:

Table 9. Options for the WSD and vector construction parameters.

Parameter	Option	Meaning	Code
Vector construction strategy	concepts only	Vector only consists of all synsets in a transcript	only
	concepts + words	Vector contains all synsets and words in the transcript	comb
	concepts replacing words	Vector contains all synsets plus words that cannot find synset match	repl
WSD strategy	1st sense	Given a word, return the 1st sense of a word as its synset	1st
	POS	Given a word, find its sysnet based on its POS role in the sentence; if there are still multiple synset matches, then make the 1st matched synset as its synset.	pos

Table 10. ONTO-WordNet results (shading experiments using the same vector construction strategy in the same color).

	Parameter values	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
Wn1st	1st * only	0.2686	0.3478	0.4422	0.4595	0.2656
Wnpos	pos*only	0.2494	0.3281	0.4398	0.4662	0.2469
Combined (Wn1st, BOW)	1st*comb	**0.343**	**0.3653**	**0.4631**	**0.4815**	**0.3382**
Combined (Wnpos, BOW)	pos*comb	0.3323	0.3588	0.4577	0.4796	0.3272
Combined (Wn1st repl BOW)	1st*repl	0.2957	0.2403	0.4371	0.4542	0.2923
Combined (Wnpos repl BOW)	pos*repl	0.3053	0.331	0.4495	0.471	0.3018

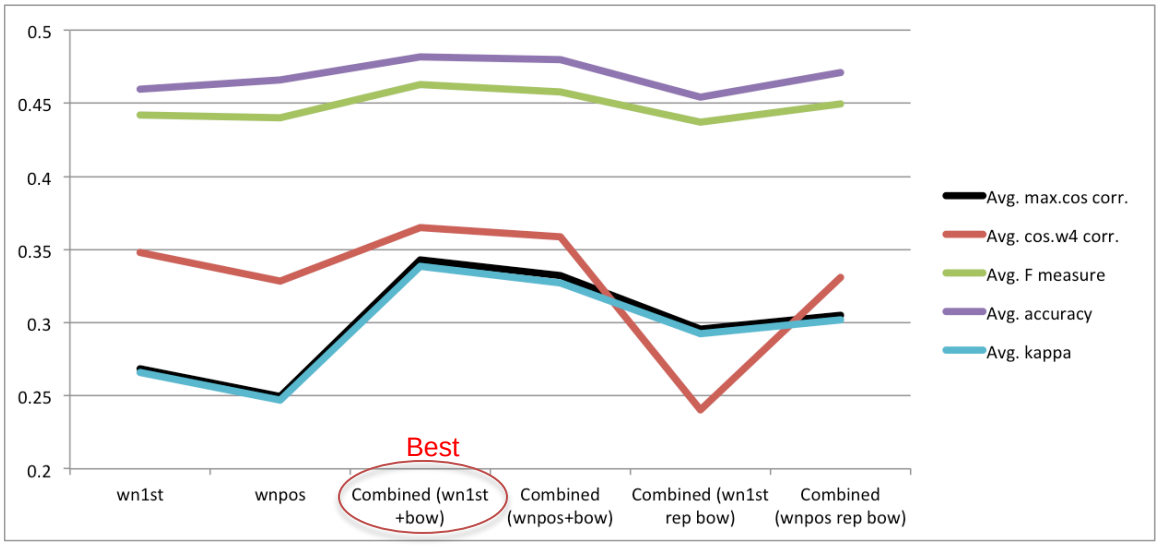


Figure 13. Visualized line chart for different ONTO-WordNet.

The results are also drawn in a line chart, in Figure 13, for visualization purposes, and the author summarizes these interpretations from the chart and tables:

1) The best performance happened in the Combined(Wn1st, BOW) experiment, which achieved the highest performance values in all measures. As reflected in Figure 13, all the performance lines reached peak values at the Combined(Wn1st, BOW) point.

2) The two experiments in the concept+words strategy, namely Combined(Wn1st, BOW) and Combined(Wnpos, BOW), outperformed the 4 experiments employing the other 2 strategies in all 5 measurement aspects. Therefore we can conclude that the concept+word option is the best of the three vector construction strategies.

3) Comparing Wn1st and Wnpos results that only use synsets in vectors, Wn1st outperformed Wnpos when measured by all measures but accuracy. It may suggest that Wn1st is a better WSD option than Wnpos for this corpus.

4) Under the *comb* vector combination strategy, Combined(Wn1st, BOW) performed better than Combiend(Wnpos, BOW) on all measurements; in contrast, under the *repl* vector combination strategy, Combined(Wn1st repl BOW) performed worse than Combined(Wnpos repl BOW) on all measurements.

5) By looking at lines in Figure 13, the author found the max.cos correlation and kappa lines almost adhered to each other. They both measure ordinal classification performance, though through different mechanisms, and exhibited consistency in measuring ordinal classes. Thus, they should be robust indicators of performance.

6) Accuracy and F measure lines also had similar trends, though in different numeric ranges. These two measures were both for nominal classification evaluation and computed from a confusion matrix, so it is probable that this correlation exists for other confusion matrices.

7) Change of parameter value had more significant effects on cos.w4 correlation, max.cos correlation, and kappa than on accuracy and F measure. In Figure 13, cos.w4 correlation, max.cos correlation, and kappa have “rocky” lines while accuracy and F

measure lines are more smooth. The author attributes their different reactions to parameter change to the different mechanisms of these evaluation measures. The first measurement group (cos.w4 correlation, max.cos correlation, and kappa) considers the classes as ordinal values, whereas the second group (accuracy and F measure) treats classes as nominal and thus loses ordinal information in their results.

Best Parameter Option (concepts+words, 1st sense)

As discussed, the best performance overall of the ONTO-WordNet experiments took place in the Combined(Wn1st,BOW) experiment, of which the parameter values were:

VectorConstruction strategy=concepts+words, *WSD strategy*=1st sense.

4.2.4 ONTO-Wikipedia parameters

The parameter concerning concept matching method was experimentally tested in ONTO-Wikipedia representation; options and performances for which are shown in Table 11 and Table 12 respectively.

Table 11. Parameter options and meanings for ONTO-Wikipedia.

Parameter	Option	Meaning
Concept matching	Direct matching	Identify candidate Wikipedia concepts from transcript text
	ESA (indirect matching)	First obtain a word to Wikipedia concepts matrix from the Wikipedia corpus; and given a transcript, compute its vector of Wikipedia concepts based on the matrix.

Table 12. Performances on different ONTO-Wikipedia parameter setups.

	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
DirectWiki	**0.1444**	**0.187**	**0.3489**	**0.3749**	**0.1366**
ESA (n=10)	0.0469	0.0483	0.3023	0.3302	0.0413
ESA (n=20)	0.0565	0.05	0.3199	0.3429	0.0515
ESA (n=50)	0.0789	0.0548	0.345	0.3583	0.075
ESA (n=100)	0.0467	0.051	0.3391	0.3606	0.0431
ESA (n=1000)	0.0183	0.0503	0.3139	0.3576	0.0156

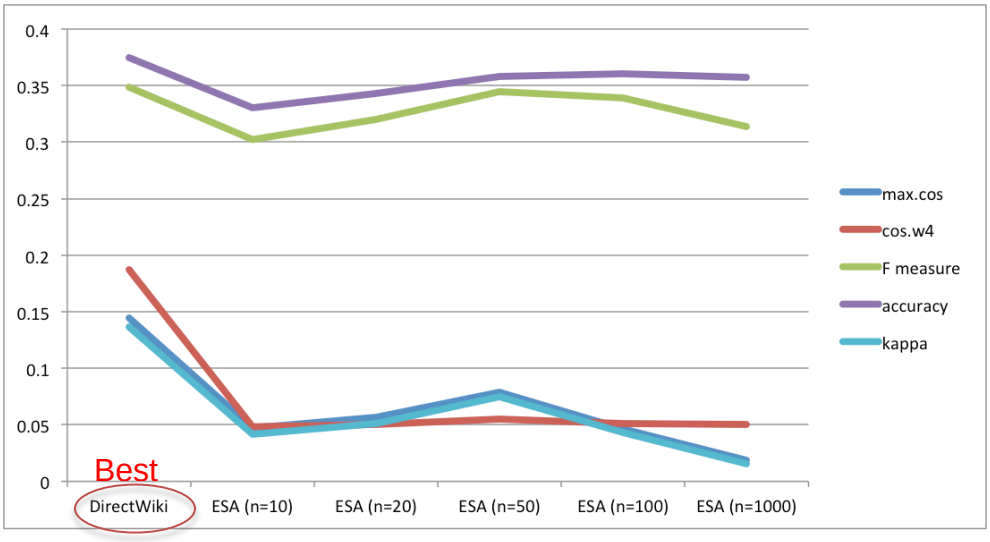


Figure 14. Performance chart for the ONTO-Wikipedia experiments.

The author arrives at the following findings by analyzing performance results in Table 12 and chart lines in Figure 14:

1) DirectWiki performed better than any ESA experiment. It is obvious that all performance measures achieved peak values at DirectWiki from the chart lines.

2) The ESA experiments resulted in fairly low performance. The 5 ESA experiments, with different dimension cutoff thresholds, exhibited low performance compared to the DirectWiki strategy, especially on ordinal class measurements. For example, for the max.cos correlation measure, the ESA experiments fell in the range of [0.0183, 0.0789], which was a much lower interval than the 0.1444 value of DirectWiki.

3) Upon inspecting vectors generated from ESA representation, Wikipedia concepts in the vectors were not so relevant to the prompt topics. Table 13 lists the top 20 Wikipedia concepts associated with prompt 099 topic (animal domestication) output from the ESA algorithm. We can find that these Wikipedia concepts were not closely related to the animal domestication topic and, thus, irrelevant vectors may cause poor performance of ESA.

Table 13. Top 20 Wikipedia concepts in the ESA vector of score level 4, prompt 099.

Rank	Wikipedia Concept	Weight
1	Antelope_Acres,_California	0.0145
2	Saskatchewan_Highway_317	0.0145
3	Grant_Township,_Antelope_County,_Nebraska	0.0134
4	Lincoln_Township,_Antelope_County,_Nebraska	0.0133
5	Royal_Township,_Antelope_County,_Nebraska	0.0133
6	Cedar_Township,_Antelope_County,_Nebraska	0.0132
7	Sherman_Township,_Antelope_County,_Nebraska	0.0132
8	Eden_Township,_Antelope_County,_Nebraska	0.0132
9	Arthur_B._Ripley_Desert_Woodland_State_Park	0.0132
10	Crawford_Township,_Antelope_County,_Nebraska	0.0131
11	Willow_Township,_Antelope_County,_Nebraska	0.0131
12	Elm_Township,_Antelope_County,_Nebraska	0.0131
13	Stanton_Township,_Antelope_County,_Nebraska	0.0131
14	Blaine_Township,_Antelope_County,_Nebraska	0.0130
15	Burnett_Township,_Antelope_County,_Nebraska	0.0130
16	Ord_Township,_Antelope_County,_Nebraska	0.0130
17	Sable_Antelope	0.0130
18	Elgin_Township,_Antelope_County,_Nebraska	0.0130
19	Custer_Township,_Antelope_County,_Nebraska	0.0129
20	Logan_Township,_Antelope_County,_Nebraska	0.0128

4) Continuing from point 3, these irrelevant concepts were possibly ranked high because the text of these concepts are short and contains a high frequency of “antelope”, which is an important word of prompt 099. For example, concept “Antelope_Acres,_California”, ranked 1st in the ESA dimensions, has only 252 words, of which 6 are “antelope”. The Wikipedia concept “Antelope”, the true concept about antelope, was in the ESA vector but was ranked lower, at 72th in the list.

Best Parameter Option (DirectWiki)

The author selected the DirectWiki method as the best parameter due to the poor performance of ESA.

4.2.5 OntoReason-WordNet Parameters

The two reasoning approaches, i.e. OntoReason-WordNet and OntoReason-Wikipedia, tackled the unknown term problem. The OntoReason-WordNet approach tuned WSD and concept similarity parameters as outlined in Table 14 below.

Table 14. Parameter options of the OntoReason-WordNet approach.

Parameter	Option	Meaning
WSD strategy	1st sense (1^{st})	Given a word, return the 1st sense of a word as its synset.
	POS (pos)	Given a word, find its sysnet based on its POS role in the sentence; if there are still multiple synset matches, then make the 1st matched synset as its synset.
Concept similarity	Path similarity (Path)	Similarity is the length of path between 2 concepts in WordNet; unknown concept weight is the average weight of its 5 most similar concepts.
	Lin similarity (Lin)	Similarity is computed based on WordNet structure and word probability from external corpus; unknown concept weight is the average weight of its 5 most similar concepts.
	Default similarity (Dft)	Assuming the unknown word has same similarities with each known concept; unknown concept weight is the average weight of all the known concepts.

The experiment results are listed in Table 15.

Table 15. Performance results of the OntoReason-WordNet experiments.

	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
WNreasoning (Wn1st, Path)	**0.2511**	0.372	**0.4342**	**0.4374**	**0.2486**
WNreasoning (Wn1st, Lin)	0.2266	0.3769	0.4241	0.4265	0.2236
WNreasoning (Wn1st, Dft)	0.2422	**0.3864**	0.4246	0.4249	0.2381
WNreasoning (Wnpos, Path)	0.2153	0.3543	0.4213	0.4253	0.2123
WNreasoning (Wnpos, Lin)	0.2119	0.3667	0.4135	0.4155	0.2076
WNreasoning (Wnpos, Dft)	0.2176	0.3709	0.412	0.413	0.2138

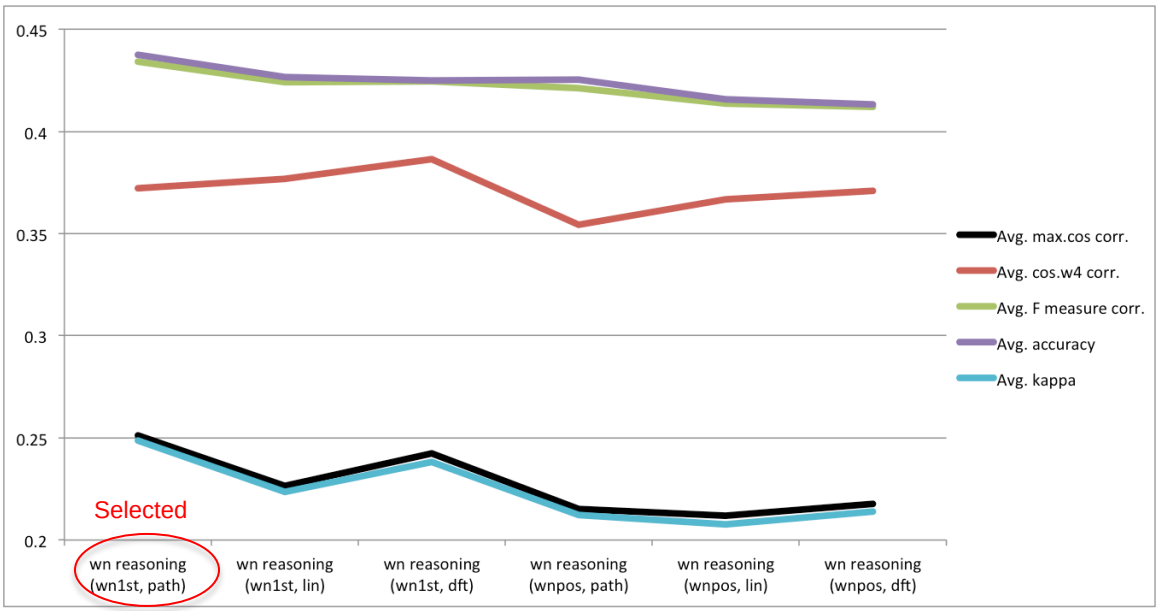


Figure 15. Performance chart of WordNet-reasoning experiments.

The author summarizes these points from the results:

1) max.cos correlations were low in all the OntoReason-WordNet experiments, regardless of parameter setup. The values were in the [0.2119,0.2511] range, lower than its [0.2494, 0.343] range in the ONTO-WordNet approach.

2) The cos.w4 correlation line appeared to be less wavy (Figure 15) than in the ONTO-WordNet approach (Figure 13), meaning it became less sensitive to parameter change in the OntoReason-Wordnet approach.

3) The max.cos correlation and kappa lines had similar trends, and the F measure and accuracy lines also highly correlate with each other. This situation also happened in ONTO-WordNet.

4) When the WSD strategy was fixed to Wn1st, for the accuracy measure, the performance rank was Path > Lin > Dft; the rank was the same when using the Wnpos option.

5) Converse to accuracy, for the cos.w4 correlation measure, when the WSD strategy was fixed to Wn1st or Wnpos, the performance rank was Path < Lin < Dft.

6) Cos.w4 correlation measure was relatively high in this approach. It was higher than the max.cos correlation in all the 6 experiments, suggesting that reasoning approach can enhance cos.w4 correlations.

Best Parameter Option (WNreasoning(Wn1st, Path))

From points 4 and 5 above, OntoReason-WordNet did not behave consistently on the accuracy and cos.w4 correlation measurements on which the ranking orders of reasoning methods were totally reverse to each other. The author chose WNreasoning(Wn1st,Path) as the optimized parameter because it reached highest value on 4 out of 5 measurements except on cos.w4 correlation while its cos.w4 correlation was relatively high too.

4.2.6 OntoReason-Wikipedia Parameters

This approach had one parameter, the concept similarity method, for approximating the weight of unknown concepts in score level vectors and test vectors.

Table 16. Parameter Options of OntoReason-Wikipedia.

Parameter	Option	Meaning
Concept similarity	Content based similarity (WikiReasoning(Content))	Compute weight of an unknown concept by averaging weights of its similar concepts (n=5); concept similarity is computed based on cosine similarity between text description of Wikipedia concepts.
	Default similarity (WikiReasoning(Dft))	Assuming the unknown word has same similarities with each known concept; unknown concept weight is the average weight of all the known concepts.

The experiment results are listed in Table 17.

Table 17. OntoReason-Wikipedia results.

	Avg. max.cos corr.	**Avg. cos.w4 corr.**	**Avg. F measure**	**Avg. accuracy**	**Avg. kappa**
WikiReasoning(Content)	0.1217	0.1929	**0.3336**	**0.3469**	0.1124
WikiReasoning(Dft)	**0.1343**	**0.1958**	0.3246	0.3413	**0.1245**

Some observations are:

1) The performance of WikiReasoning(Content) and WikiReasoning(Dft) were close on all 5 measurements, though WikiReasoning(Content) was slightly higher than WikiContent(Dft) on F measure and accuracy and slightly lower than WikiContent(Dft) on the other 3 measurements.

2) Continuing from point 1, WikiReasoning(Content) outperformed WikiReasoning(Dft) on the three ordinal class measurements, while WikiReasoning(Dft) outperformed WikiReasoning(Content) on the two general machine learning measurements.

3) Content based similarity for unknown concepts did not perform better than default similarity. Though theoretically sound, in the experiments, it did not improve over the default similarity option.

Best Parameter Option (WikiReasoning(Content))

It turned out that no one option outperforms the other one all the time. OntoReason-Wikipedia performances were relatively low compared to the OntoReason-WordNet case. The author selected WikiReasoning(Content) as the best parameter since content-based similarity was more a meaningful way to reason weights of unknown concepts than default similarity.

4.3 Hypothesis Analysis (between-approach analysis)

Similar to parameter analysis, the author focused on comparing the average performance of the approaches. Additionally, since every representation approach had some parameter options, the author primarily sought to identify the parameter option achieving the best performance and uses best performance to compare between approaches. For example, BOW(tfidf) and LSA(k=40) are the best parameter options in BOW and LSA approaches respectively, and they are used for between-group analysis when comparing BOW and LSA.

4.3.1 BOW vs. LSA (H1)

H1*. Content scoring models from LSA representation outperform content scoring models from BOW representation in predicting speaking proficiency.*

Table 18. BOW and LSA results.

	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
BOW(tfidf)	**0.3494**	**0.3556**	**0.4627**	**0.4786**	**0.3441**
LSA(k=40)	0.2506	0.151	0.3931	0.4442	0.2393

This is the comparison between the two baseline systems. We can see that BOW(tfidf) exceeded LSA(k=40) in all measurement aspects. The author considers these factors as contributing to LSA's inferior performance:

1) Data sets for generating LSA space were small. Prompts 098, 099, 100, and 101 used 212, 204, 206, and 202 training transcripts to generate LSA vector space, respectively. Since LSA learns from word co-occurrence, small data size may result in a distorted co-occurrence matrix, from which distorted latent concepts were derived.

2) LSA tended to eliminate similarity discrepancy. The scoring model cosine similarity-based, which predicts scores based on the level with the highest similarity to

the test transcript. It seems that LSA based similarities tended to be numerically closer; in other words, given a test vector, its similarities with score levels 2, 3, and 4 (sim2, sim3, sim4 henceforth) were numerically closer than BOW based similarities. Taking prompt 100 as an example, the average discrepancy between sim3 and sim2 (sim3 – sim2) was 0.0167 on LSA and is 0.0284 on BOW, indicating that using LSA shrank similarity discrepancy in this study. As we know, one potential harm was that similarities are close to each other, making it harder to discern the closest score level, given a test transcript.

Response to the hypothesis

The hypothesis was not supported by the test results, which on the contraty showed that BOW outperformed LSA on all measurements.

4.3.2 BOW vs. ONTO (H2)

H2*. Content scoring models from ONTO representation outperform content scoring models from BOW representation in predicting speaking proficiency.*

Table 19. BOW and ONTO-WordNet performance.

	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
BOW(tfidf)	**0.3494**	**0.3556**	**0.4627**	**0.4786**	**0.3441**
Wn1st	0.2686	0.3478	0.4422	0.4595	0.2656
Wnpos	0.2494	0.3281	0.4398	0.4662	0.2469
DirectWiki	0.1444	0.187	0.3489	0.3749	0.1366

This table shows that only using WordNet synset in vectors (Wn1st and Wnpos) resulted in worse performance than BOW(tfidf), in all measurement aspects. Comparing within the WordNet representation, Wn1st had better performance than Wnpos on all measurements except accuracy.

It is an interesting result that Wn1st performed less competently than BOW, though Wn1st representation was sound theoretically. The author summarizes these possible reasons:

1) The strength of WordNet synsets, namely synonym grouping or dimensionality reduction, may not be released to full extent. A WordNet synset subsumes one or several words and, thus, makes vector representation more compact. However, when corpus and document sizes are small, there is a high chance that synonymous words do not all occur in the corpus or documents. In the worst scenario, no synonymous words are present in the corpus, so that each single word is matched to a distinct synset with no synonymous words grouped as one dimension. In this sense, WordNet is similar to BOW representation –words are simply labeled as distinct synset. Since this study used a small corpus, there were many fewer chances to group synonymous words in this corpus than in a huge corpus. The situation of only a handful of synonyms being merged to one synset dimension restrained the benefit of reducing dimensionality brought by WordNet.

Let's look at some statistics about dimensions (all from training set of the 1st run in the 3-fold cross-validation, more specifically, the score level 4 vector). For each prompt, the number of vector dimensions was reduced when representation is changed from BOW(tfidf) to Wn1st.

Table 20. Number of vector dimensions, score level 4.

	Prompt 098	**Prompt 099**	**Prompt 100**	**Prompt 101**
Num of dimension in BOW(tfidf)	615	478	682	617
Num of dimensions in Wn1st	474	376	537	464
Num of dimensions in DirectWiki	26	32	26	43

Table 20 shows that using WordNet led to dimensionality reduction to some extent. Besides the words that did not have a match in WordNet (e.g. articles and prepositions), dimensionality was further reduced by grouping synonyms to synsets. For example, "*level*" and "*degree*" in prompt 098, "*choose*" and "*select*" in prompt 099, "*totally*" and "*completely*" in prompt 100, and "*persons*" and "*individual*" in prompt 101.

However, if a word's synonym does not occur in the corpus, the opportunity to take advantage of merging synonymous words is lost. For example, "*requirement*" and "*demand*" are synonyms, but only "*requirement*" is present in the corpus, and thus it has no chance to be combined to a dimension with "*demand*".

It seems that reducing dimensions alone did not improve performance over BOW(tfidf) but lowered the performance instead. However, as discussed above, synonym grouping did not occur for many words in this study and, thus, we need more evidence to support or deny that using synsets causes performance to drop.

2) Level of synset lookup may affect resulting synset vector. The current ONTO-WordNet method finds the matched synset, given a word and its sense information, while it does not consider other related synsets, such as its hypernym (parent synset). Hypernyms of the matched synsets may be important for representation because they are higher-level concepts with more generality so that a document can be better associated to another potentially relevant document by explicitly including more hypernyms. Hotho et al. (2003a) illustrates usage of hypernyms for "beef" and "pork" that share a common parent, "meat".

Here the author uses words "*chance*" and "*possibility*" from the data set for a similar demonstration (Figure 16). For example, word "*chance*" appears in transcript

7588019-VB531101 and its 1st sense synset ID is *{SID-14483917-N}* [5,6,7,8]. The sentence is:

> "*Automobile uh gave uh the people the **chance** to uh visit other parts of the country…*"

This synset is subsumed by its hypernym, *{SID-05951180-N}*, which considers possibility. A sentence that contains word "*possibility*" is in transcript 7527510-VB531101:

> "*In this way uh way they have the **possibilities** to see different part of the country …"*

The first transcript is semantically associated with the second one via the parent-child relation between the "*possibility*" and "*chance*" synsets. However, since current approaches do not include hypernyms of synsets in the vector, the transcripts cannot be associated in this way and therefore their distance may be underestimated. On the other hand, though including hypernym synsets may enhance similarity between two transcripts, it could also diminish similarity by introducing an overflow of general synsets. Hotho et al.'s (2003) study suggests including synset hypernyms for up to 5 levels for best performance, and since this experiment only contains matched synset in vector, this could partly account for the low ONTO-WordNet performance.

---

[5] Synset can be mentioned by either ID or by label.

[6] Synset ID varies in different WordNet interfaces, such as the JWI Java WordNet Interface and the online WordNet interface (http://wordnetweb.princeton.edu/perl/webwn). For example, given word "chance", the synset ID of its 1st sense in JWI is {SID-14483917-N}, and its ID in the online interface is {14507501}. However, they share the same synset key, and in this example, the synset key is chance%1:26:00::.

[7] To be consistent, when mentioning a synset by ID, the author uses the ID from JWI.

[8] When mentioning a synset by its label, the author follows naming convention of the NLTK package, which labels synsets in the way of *<lemma>.<pos>.<number>* (Bird, Klein, & Loper., 2009). For example, the 1st sense of "chance" is labeled as *chance.n.01*.

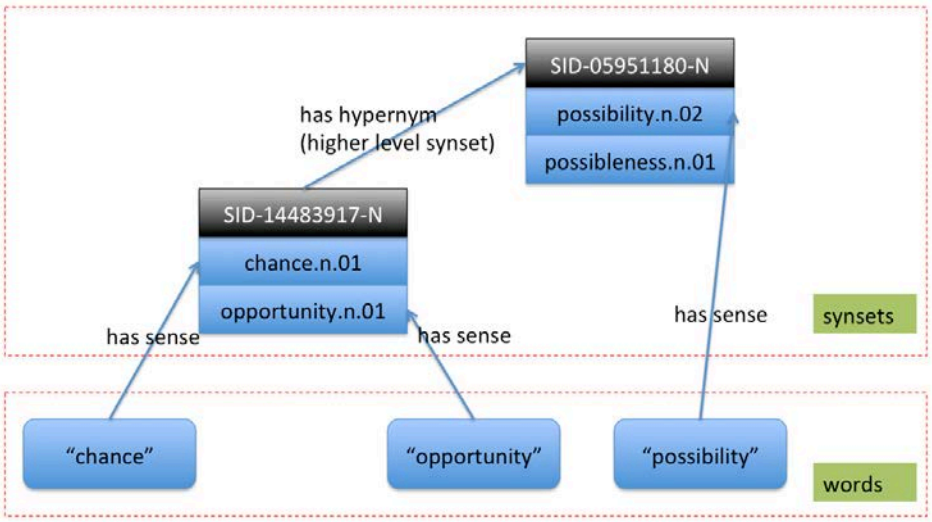


Figure 16. Word, synsets, and hypernym of a synset.

3) Multi-word expression may also contribute to poor performance. In this approach, only single words were matched to WordNet synsets, whereas multi-word expressions were ignored. Some WordNet synsets subsume phrases, e.g. "*big cat*" and "*open university*". Only matching single words can lead to imprecise matching results, for example, "big cat" should be mapped to one synset *{big cat.n.01}*, instead of being split into "*big*" and "*cat*" and then mapped to *{big.n.01}* and *{cat.n.01}* respectively. This deficiency may potentially be complemented by the DirectWiki approach that can locate multi-word expressions from text.

4) Synset matching errors can bring noise to vectors. Stopwords such as "a" and "I" are mapped to a synset in Wn1st, however they actually are not included in WordNet (more details in section 4.4.1. Wn1st finds their match because they are the acronym of "angstrom" and "Iodine" respectively. Including such incorrect synsets brought noise to the representation and may further lower the scoring performance.

Besides Onto-WordNet, Onto-Wiki (DirectWiki) also performed less well than BOW, and the author summarizes the reasons for its poor performance:

1) The vector length was very short. DirectWiki located many fewer concepts from transcripts than Onto-WordNet. One DirectWiki vector contains 3.84 concepts, while one BOW and Wn1st vectors contains 44.62 and 46.31 dimensions on average respectively. Shorter vectors contain less information, making it difficult to distinguish between transcripts of different qualities through similarity measurement. As listed in Table 20, the number of dimensions for score level 4 vector is in the range of [26, 43], much lower than Wn1st and BOW. Therefore the DirectWiki vector was not a good representation for a transcript.

2) Some transcripts had no Wikipedia concept match. The Wikifier package that was used to map text to Wikipedia concepts was based on a global coherence measurement instead of simply string match. It had the advantage of disambiguating concepts of the same string form, but usually resulted in fewer concepts than string match. Sometimes it even resulted in outputting no concept match. For example, 7586861-VB531100 is a score 4 transcript, but no Wikipedia concept was matched to its text. Taking its 1st sentence as example:

"*The woman is worried uh about uh her uh schoolwork*."

"Woman" is actually a Wikipedia concept but was not returned by Wikifier. When no match was found, the program returned an empty vector, which has 0 similarity with any score level vectors. This of course made it impossible to find the most similar score level vector.

3) DirectWiki can find some phrases (multi-word expressions), but in a small amount. The advantage of DirectWiki over Wn1st is that it can locate phrases, which

are then matched to Wikipedia concepts. For example, transcript 7542960-VB531100 contains phrase “time management”:

“ … Actually this is the *time management* she doesn’t know …”

However, the number of identified phrases was small, e.g. only 1 phrase was found in this transcript. The vector representation can be enriched if more phrase-based concepts are returned.

4) Same as Wn1st, errors in Wikipedia concept matching accounted for DirectWiki’s poor performance. In transcript 7409857-VB531100, “energy” was incorrectly mapped to Wikipedia concept “*Energy_and_society*”, while its correct concept is actually the “*Energy*” page in Wikipedia. There were also errors in recognizing phrase-based concepts. For example, transcript 7605166-VB531100 contains “*extra time*”, which however was incorrectly matched to “*Overtime (sports)*” in Wikipedia.

5) Wikifier toolkit did not always return the same Wikipedia concepts given the same word. Wikifier employs local context and global concept coherence to determine concept match (Ratinov, Roth, Downey, Anderson, 2011), and therefore the same surface word in different transcripts may result in being mapped to different concepts. For example, word “*math*” was mapped to the “*Mathematics*” concept (correct) in transcript 7605166-VB531100 whereas in transcript 7508663-VB531100 it was mapped to the “*Mathematics education*” (incorrect). In fact, a word is usually subsumed by one concept in this data set; the other mapped concepts are usually wrong, such as the “*Mathematics education*” concept in this example. The wrongly identified concepts

literally added unnecessary dimensions and bring noise to the DirectWiki representation.

Response to the hypothesis

Within this study, empirical evaluation suggested that only using WordNet synsets or Wikipedia concepts for representation had adverse effect on speech scoring performance; however, combining synsets and word vectors can enhance performance over the BOW baseline. Additional effects of combining vectors are discussed in section 4.3.5.

4.3.3. LSA vs. ONTO (H3)

H3. *Content scoring models from ONTO representation outperform content scoring models from LSA representation in predicting speaking proficiency.*

Table 21. LSA and ONTO results.

	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
LSA(k=40)	0.2506	0.151	0.3931	0.4442	0.2393
ONTO-WordNet (Wn1st)	0.2686	0.3478	0.4422	0.4595	0.2656
ONTO-Wiki (DirectWiki)	0.1444	0.187	0.3489	0.3749	0.1366
ONTO-WordNet (Combined(Wn1st, BOW))	**0.343**	**0.3653**	**0.4631**	**0.4815**	**0.3382**

This hypothesis compares effects of latent concept and explicit concept representations. From the comparison table, we can see that both ONTO-WordNet experiments (Wn1st and Combined(Wn1st,BOW)) achieved higher performance than LSA on all measurements. However, LSA outperformed ONTO-Wiki (DirectWiki) on all measurements but cos.w4 correlation. These results partially support the hypothesis, and more specifically, using ONTO-WordNet can outperform LSA but ONTO-Wiki cannot.

Again, the author thinks the unsatisfactory performance of LSA is attributable to the small amount of training data, making the generated vectors not suitable for distinguishing different speech score levels. The failure of DirectWiki is primarily due to its small number of identified concepts.

Response to the hypothesis

In this study, the ONTO-WordNet representations, as either vectors of synsets or combinations of synsets and words, outperformed LSA. The experimental results support the hypothesis when the ontology in use is WordNet but challenge the hypothesis when ontology is Wikipedia.

It is noteworthy that this response is limited by the data set size. Foltz et al. (1999) employ LSA for essay scoring and acquire correlation (similar to the max.cos correlation measure) as high of 0.701 but their LSA subspace is trained from external corpus instead of the 1205 essays in local corpus. Though Foltz et al. (1999) do not mention size of their training corpus, it can be conjectured that using external corpus relates to its local corpus size, as 1205 documents is still a small corpus. Bradford (2008) recommends LSA dimension cutoff should be between k=200 and k=500, and the poor performance of LSA(k=200) also reflected the fact that the data size is inappropriate for LSA training.

4.3.4 ONTO vs. OntoReason (H4)

H4*. Content scoring models from OntoReason representation have better predictiveness on speaking proficiency than the content scoring models from ONTO representation.*

The author analyzed results of ONTO and OntoReason in WordNet and Wikipedia groups, respectively. In the first group, ONTO-WordNet and OntoReason-WordNet were compared. Comparing Wn1st and WNreasoning(Wn1st,Path), she found WNreasoning(Wn1st,Path) only improved over Wn1st on the cos.w4 correlation measurement. Comparing WNreasoning(Wn1st,Dft) with Combined(Wn1st,BOW), the observation was the same: only cos.w4 correlation was improved when using reasoning.

Table 22. The WordNet group for ONTO and OntoReason comparison.

	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
ONTO-WordNet (Wn1st)	0.2686	0.3478	0.4422	0.4595	0.2656
ONTO-WordNet (Combined(Wn1st, BOW))	**0.343**	0.3653	**0.4631**	**0.4815**	**0.3382**
OntoReason-WordNet (WNreasoning(Wn1st, Path))	0.2511	**0.372**	0.4342	0.4374	0.2486

The second group compared ONTO-Wikipedia and OntoReason-Wikipedia. The observation was similar to the WordNet group, OntoReason-Wikipeida outperformed DirectWiki when measured by cos.w4 correlation, but was inferior to DirectWiki on other measures.

Table 23. The Wikipedia group for ONTO and OntoReason comparison.

	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure corr.	Avg. accuracy	Avg. kappa
ONTO-Wiki (DirectWiki)	**0.1444**	0.187	**0.3489**	**0.3749**	**0.1366**
OntoReason-Wikipedia (WikiReasoning(Content))	0.1217	**0.1929**	0.3336	0.3469	0.1124

Response to the hypothesis

The author proposes to partially accept the hypothesis. The hypothesis is valid under certain circumstances: both OntoReason-WordNet and OntoReason-Wikipedia

improved the cos.w4 correlation performance but fail edon other performance measures, compared to the non-reasoning approaches.

4.3.5 Combination Effects

The above displayed results show that ONTO and Onto-Reason approaches had lower performance than BOW(tfidf) on most measurements, except for several sporadic cases. However, combining WordNet and BOW vectors can sometimes improve performance, e.g. Combined(Wn1st,BOW). Therefore the author gathered results from combining different types of vectors here (word, synset, and Wikipedia concept vectors) and analyzed what effects the combined vectors had on automatic scoring performance.

A given document can be represented by three vector types, namely BOWVec for word vector, SynVec for synset vector, and WikiVec for Wikipedia concept vector. The vectors are written as [w_1, …, w_k], [syn_1, …, syn_m], and [$wiki_1$, …, $wiki_n$], respectively. The combined vector is therefore

[w_1, ..., w_k, syn_1, ..., syn_m, $wiki_1$, ..., $wiki_n$]

Given two combined vectors, CV_a and CV_b, whose vector values are

$[a_{w_1}, \dots, a_{w_k}, a_{syn_1}, \dots, a_{syn_m}, a_{wiki_1}, \dots, a_{wiki_n}]$ and

$[b_{w_1}, \dots, b_{w_k}, b_{syn_1}, \dots, b_{syn_m}, b_{wiki_1}, \dots, b_{wiki_n}]$ respectively.

The combined vectors are composed of the three types of vectors: the values for the BOWVec chunk of CV_a is $[a_{w_1}, \dots, a_{w_k}]$ part, the valule of the SynVec chunk for CV_a is $[a_{syn_1}, \dots, a_{syn_m}]$, and the values of the WikiVec chunk for CV_a is the $[a_{wiki_1}, \dots, a_{wiki_n}]$.

The cosine similarity between CV_a and CV_b is

$$CosSim\,(CV_a, CV_b) = \frac{\sum_{i=1}^{k} a_{w_i} * b_{w_i} + \sum_{j=1}^{m} a_{w_j} * b_{w_j} + \sum_{h=1}^{n} a_{w_h} * b_{w_h}}{|CV_a| * |CV_b|}$$

Because BOWVec, SynVec, WikiVec are normalized to length of 1, $|CV_a| * |CV_b|$ results in $\sqrt{3} * \sqrt{3} = 3$; also because of the normalized vectors, $CosSim\ (BOWVec_a, BOWVec_b) = \sum_{i=1}^{k} a_{w_i} * b_{w_i}$, $CosSim\ (SynVec_a, SynVec_b) = \sum_{j=1}^{m} a_{w_j} * b_{w_j}$, and $CosSim\ (WikiVec_a, WikiVec_b) = \sum_{h=1}^{n} a_{w_h} * b_{w_h}$. Therefore we derive the following equation:

$$CosSim\ (CV_a, CV_b) = {}^{1}/_{3}\ CosSim\ (BOWVec_a, BOWVec_b) + {}^{1}/_{3}\ CosSim\ (SynVec_a, SynVec_b) + {}^{1}/_{3}\ CosSim\ (WikiVec_a, WikiVec_b)$$

which means the cosine similarity between two combined vectors equals to the average of their BOWVec similarity, SynVec similarity, and WikiVec similarity.

The above equation implies that the three similarities hold the same importance, 1/3. We can also assign different importance to the similarities if we think one particular similarity, e.g. the BOWVec similarity, has more importance. The author thus multiplies the three similarities with α, β, γ respectively, which indicate their relative importance in the total similarity. These importance, α, β, γ, are called “importance multiplier” here. Then the similarity equation becomes:

$$CosSim\ (CV_a, CV_b) = \alpha * CosSim\ (BOWVec_a, BOWVec_b) + \beta * CosSim\ (SynVec_a, SynVec_b) + \gamma * CosSim\ (WikiVec_a, WikiVec_b) \quad (\alpha + \beta + \gamma = 1)$$

Table 24 lists performance of combined vectors in which each vector type has the same similarity importance. That is to say, the vectors shared the same importance multiplier for similarity. For example, for Combined (Wnpos, BOW), the importance multiplier values are α=0.5 and β=0.5, which do not reflect difference in similarity importance. In Table 25, combined vectors have different importance. For example, Combined (BOW=0.7, Wn1st=0.3) means α=0.7 and β=0.3.

Table 24. Results of combined vectors, where vectors share the same importance multiplier. (shaded cells means it is the highest value among all approaches, for a particular measurement)

	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
BOW(tfidf)	0.3494	0.3556	0.4627	0.4786	0.3441
Combined(Wn1st, BOW)	0.343	**0.3653**	0.4631	**0.4815**	0.3382
Combined (Wnpos, BOW)	0.3323	0.3588	0.4577	0.4796	0.3272
Combined (BOW, Wiki)	0.1923	0.2696	0.3771	0.4038	0.1854
Combined (Wn1st, Wiki)	0.1658	0.2758	0.3706	0.3959	0.1608
Combined (Wnpos, Wiki)	0.1713	0.2702	0.3696	0.3951	0.1661
Combined (BOW, Wn1st, Wiki)	0.2268	0.3244	0.3978	0.4203	0.2204
Combined (BOW, Wnpos, Wiki)	0.2154	0.3201	0.3875	0.4129	0.2086
Combined (BOW, esa10)	**0.3582**	0.2895	**0.4682**	0.4789	**0.3548**
Combined (BOW, esa20)	0.3245	0.2627	0.4516	0.4608	0.3209
Combined (BOW, esa50)	0.3431	0.1935	0.4587	0.4675	0.3342
Combined (BOW, esa100)	0.3282	0.1304	0.4324	0.4478	0.3075
Combined (BOW, esa1000)	0.2278	-0.0699	0.3469	0.3778	0.1791
Combined (Wn1st repl BOW)	0.2957	0.2403	0.4371	0.4542	0.2923
Combined (Wnpos repl BOW)	0.3053	0.331	0.4495	0.471	0.3018
Combined (Wiki repl BOW)	0.1493	0.2264	0.3609	0.3958	0.1451

When vector types share equal importance, the results show that the highest value for every measurement occurs in combination approaches, except for the max.cos correlation measurement whose highest value is in BOW(tfidf). It indicates that combining vectors in an additive way may enhance performance. Combined(Wn1st, BOW) is a good example of combining word and synset vectors, which achieves the highest cos.w4 correlation and accuracy values in Table 24.

The author further explored effects of assigning different importance multiplier to the three vector types (in Table 25). She concluded with the following points from Table 25:

Table 25. Results of combined vectors with different importance multipliers. (Shaded cells are the highest values in this table; the last row lists the highest values from Table 24).

	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
BOW(tfidf)	**0.3494**	0.3556	0.4627	0.4786	0.3441
Combined (BOW=0.7, Wn1st=0.3)	**0.3704**	0.3651	**0.4710**	0.4869	**0.3648**
Combined (BOW=0.75, Wn1st=0.25)	0.3699	0.3643	0.4709	0.4860	0.3642
Combined (BOW=0.6, Wn1st=0.4)	0.3634	0.3658	0.4690	0.4863	0.3583
Combined (BOW=0.7, Wn1st=0.2, wiki=0.1)	0.3511	**0.3797**	0.4667	0.4851	0.3464
Combined (BOW=0.7, Wn1st=0.25, wiki=0.05)	0.3605	0.3773	0.4699	**0.4877**	0.3552
Combined (BOW=0.6, Wn1st=0.2, wiki=0.2)	0.3100	0.3638	0.4401	0.4607	0.3044
Combined (BOW=0.7, Wn1st=0.1, wiki=0.2)	0.3061	0.3613	0.4394	0.4591	0.3007
Highest from Table 24. (combined vector with equal importance multiplier)	0.3582	0.3653	0.4682	0.4815	0.3548

1) It turned out that the highest values were all in combined approaches (the green shaded cells). The performance was especially good on Combined(BOW=0.7, Wn1st=0.3), which reached highest values on three measurements. It seemed that when combining BOW and Wn1st in an appropriate ratio, we can achieve better performance than the BOW baseline on all measurements.

2) As we can see, BOW still played an important role in the combined vector. For example, in Combined(BOW=0.7, Wn1st=0.3) it made 70% of the overall similarity measurement. Performance went down when BOW importance was reduced in the author's other experiments.

3) Adding WikiVec to the combined vector and adjusting its importance multiplier did not improve performance of combined vectors much. When WikiVec importance multiplier γ increased, such as in Combined (BOW=0.6, Wn1st=0.2, wiki=0.2) and

Combined (BOW=0.7, Wn1st=0.1, wiki=0.2), performance tended to drop on all measurements. Further evidence was that the best combination Combined(BOW=0.7, Wn1st=0.3) contained no WikiVec.

4.4 In-Depth Analysis

Due to the central role of document vectors in this study, the author selected some sample transcripts and manually inspected their vectors to facilitate performance analysis. Vectors generated from various representation approaches were outputted as a list of terms to facilitate the inspection.

The table below displays corpus and document sizes under different representations:

Table 26. Size of document and vocabulary from different representation approaches.

	BOW (term=word)	Wn1st (term=synset)	Wnpos (term=synset)	DirectWiki (term = Wikipedia concept)
Avg. size of corpus vocabulary	1288.75	851.25	854.0	81.5
Avg. num. of terms / speech response	44.62	46.31	36.89	3.84

The average size of corpus vocabulary tells how many terms are identified within each prompt's corpus on average (the 1st line of Table 26). BOW generated the largest vocabulary, 1288.75 on average, and DirectWiki produced the smallest vocabulary size, 81.5 on average. The three ontology-based representations did reduce vector dimensionality by using concept-level units.

For the average number of terms per speech response, Wn1st resulted in the largest number of distinct terms, and DirectWiki still produced the fewest terms. Unlike statistics in vocabulary size of a corpus, Wn1st had higher number of terms per

response than BOW. This shows that Wn1st can reduce the overall corpus dimensionality but at document level finds more terms than BOW.

4.4.1 Analysis of Wn1st Vectors

1) Some examples are shown below about synonymous words that were merged to one synset (for dimensionality reduction). Synonymous words, within a document or between documents, were merged to one dimension through Wn1st representation, in order to reduce overall dimensionality. Table 27 lists example synonymous words along with their source files for each prompt.

Table 27. Synonymous words in each prompt.

Prompt	Synonymous words	WordNet SynsetID
098	“result” (in transcript 7544670-VB531098), “results” (in transcript 7589423-VB531098), “effects” (in transcript 7611667-VB531098)	{*SID-11410625-N*}
099	“choose” (in transcript 7571930-VB531099), “select” (in transcript 7667147-VB531099), “choosed” (in transcript 7667232-VB531099)	{*SID-00674607-V*}
100	“meet” (in transcript 7571032-VB531100), “encounter” (in transcript 7591389-VB531100), “met” (in transcript 7655049-VB531100)	{*SID-02023107-V*}
101	“travel” (in transcript 7508663-VB531101), “go” (in transcript 7521161-VB531101), “move” (in transcript 7552081-VB531101)	{*SID-01835496-V*}

2) A fairly large number of merged synonyms shared root words. The author inspected the words under these identified synsets from corpus, and she found many are actually words with same roots. For example, synset {*SID-07357388-N*} contains words “*improvement*” and “*improvements*”, synset {*SID-00594621-V*} contains words “*knows*” and “*know*”. These synonyms are morphological variants of a same root word, whose effect can also be achieved by stemming words. It is also noteworthy that these variants possess the same word sense but stemming does not consider sense information.

3) The same word was always mapped to the same synset in Wn1st. This is also the polysemy issue, in which the same word has multiple senses. For example, "*American*" can be noun or adjective depending on the content, but it was only mapped to its 1st sense here. Obviously this sometimes caused errors. Continuing with the "*American*" example, in WordNet its 1st sense is adjective synset {*SID-02927512-A*}, but in this sentence:

- ✓ *This is good for all the American/NNP*." American is a noun here but it was matched to its 1st synset adjective {*SID-02927512-A*}.

4) Errors also happened in matching words to synsets, due to different reasons.

The first reason is that a word's correct sense does not exist in WordNet. For example, word "I" was matched to synset {*SID-14641397-N*}, a chemical element. It is because WordNet does not contain the personal pronoun "I" but only the "I" as chemical element; if we do not notify WordNet this "I" is a personal pronoun in the sentence, then the wrong match returns. If done correctly, this "I" should not be matched to anything in WordNet. This POS related error can be alleviated by using the POS method. Other such examples include:

- × "or" is matched to synset *{Oregon.n.01}* (same as {*OR.n.03}*) for the Oregon state
- × the article "a" is matched to a synset for the metric unit "angstrom" (in transcript 7597365-VB531099).

The second reason is that a word's correct sense exists in WordNet but was not the 1st sense of that word. In the sentence "The second one is that the social structure of the herd" (of transcript 7597365-VB531099), word "one" was incorrectly recognized as an adjective synset *{one.a.01}*, but actually it should be matched to its 2nd noun synset *{one.n.02}*, meaning "a single person or thing".

4.4.2 Analysis of Wnpos Vectors

1) As in the above Wn1st analysis, synonymous words were also merged to a single synset for dimensionality reduction in the Wnpos method (shown in Table 28).

Table 28. Some examples of merged synonyms.

Prompt	Synonymous words	WordNet SynsetID
098	"purpose" (in transcript 7543583-VB531098), "intent" (in transcript 7552081-VB531098)	{*SID-05982152-N*}
099	"trying" (in transcript 7530823-VB531099), "try" (in transcript 7537027-VB531099), "attempted" (in transcript 7614358-VB531099)	{*SID-02530167-V*}
100	"idea" (in transcript 7583451-VB531100), "thoughts" (in transcript 7588019-VB531100), "ideas" (in transcript 7673715-VB531100)	{*SID-05833840-N*}
101	"way" (in transcript 7527510-VB531101), "fashion" (in transcript 7543743-VB531101), "mode" (in transcript 7550690-VB531101) "style" (in transcript 7564857-VB531101)	{*SID-04928903-N*}

2) Like Wn1st, a number of identified synsets were composed of different morphological forms of the same word. It is an observation similar to Wn1st. For example, synset {*SID-00137313-V*} contains the words "*affect*" and "*affects*" that are different tenses of "affect"; synset {*SID-05898568-N*} contains the words "*program*" and "*programs*" that are single and plural forms of the word program.

3) Unlike Wn1st, the same word occurring in different sentences can be mapped to different synsets, depending on its POS roles in the sentences. For example, the word "*American*" was mapped to a noun and an adjective synsets:

- ✓ In the sentence "*American/JJ people easily travel to a small country or nearby country and there is an increased mobility for them*" (transcript 7543998-VB531101), "American" was identified as adjective synset {*SID-02927512-A*} because its POS role was adjective (JJ).
- ✓ In sentence "*This is good for all the American/NNP*" (transcript 7581194-VB531101), Wnpos matches "American" to noun synset {*SID-09738708-N*} because it was a noun (proper noun, singular).

4) Wnpos corrected some errors of Wn1st. As discussed in the above section, Wn1st erred in matching words to synsets because it used the 1st returned synset; this

error can be resolved by informing the system the POS role of the word. The errors listed in 4) of section 4.4.1 can be corrected by using Wnpos in some cases:

- ✓ “or” was labeled as CC (conjunction) by the POS tagger, and Wnpos found no match in WordNet
- ✓ “a” was recognized as DT (determiner) by the POS tagger, and thus resulted in no match.
- × “one” in the sentence “the second one is…” was correctly labeled as NN (noun), and the matched synset was *one.n.01*, meaning the number one. Here Wnpos had the correct sense for “one”, but it selected the wrong noun sense, because the Wnpos rule was that if there were multiple senses of a particular type (e.g. multiple noun senses) then it returned the 1st sense of that type.

5) POS errs in Wnpos. The errors made by POS taggers may be propagated to next steps, including matching words to synsets and possibly the scoring model. POS errors usually occurred in speech with many grammatical errors. Some POS errors are:

- × “*There's uh/JJR due uh/PRP due to the invention of automobile this increased mobility*.” (in transcript 7537027-VB531101, scored 2). This is an ungrammatical sentence, and the two “*uh*”s were incorrectly tagged (should be UH interjection). But since “*uh*” is not included in WordNet anyway, this POS error did not affect final results.
- × “*And the second one/CD is mobility*.” (in transcript 7546368-VB531101, scored 2). This sentence is grammatically correct, but the POS tagger labeled word “*one*” as CD (cardinal number), while it should be a noun. Because there is no CD sense in WordNet, this word “*one*” was not matched to any WordNet synset.

4.4.3 In-Depth Analysis of ONTO-WordNet vs. BOW

ONTO-WordNet performed less well than BOW when not combined with word vectors. This is an interesting observation because ONTO is a theoretically sound representation. Since DirectWiki (ONTO-Wikipedia) did not generate a long and good enough vector, the author focused on Wn1st of ONTO-WordNet for deeper analysis to understand the reason of its failure.

The author ran a series of side experiments to further understand why Wn1st resulted in lower performance than BOW. The differences from Wn1st and BOW experiment mechanism were: 1) Wn1st grouped synonyms; 2) Wn1st grouped words with same morphological roots; 3) Wn1st did not use stoplist while BOW does. The

three differences may all contribute to the difference between BOW and Wn1st. In order to inspect how much difference is contributed by the 1st point, we can make BOW and Wn1st has similar setups on points 2 and 3 such that the setup difference is only in point 1.

As we know, for point 2, we can do stemming on BOW to make it similar as grouping words with same roots in Wn1st, and can also keep stopwords in the BOW representation for point 3. Therefore the author first ran a BOW experiment with similar setup to Wn1st: stemming and not using stoplist.

The results in Table 29 showed BOW(stemming, non-stop) still outperformed Wn1st, but it was in a much lower performance than the BOW(tfidf) option which did no stemming and uses stopwords.

Table 29. Experiment results for understanding effects of using stopwords and merging dimensions.

	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
BOW(tfidf)	**0.3494**	**0.3556**	**0.4627**	**0.4786**	**0.3441**
BOW(stemming, non stop)	0.2938	0.3489	0.4331	0.4564	0.2882
BOW(stemming, stop)	0.3177	0.334	0.4497	0.4703	0.3115
Wn1st	0.2686	0.3478	0.4422	0.4595	0.2656
Wn1st (stop)	0.2692	0.3337	0.4415	0.4604	0.2666

It turns out BOW resulted in lower performance when combining some dimensions (due to stemming) and keeping stopwords. It is an interesting question that which one, stemming or non-stop, caused this performance drop. Therefore the author ran another experiment to examine whether using stopwords made a difference. She compared BOW(stemming, non-stop) and BOW(stemming, stop), the results of which

suggested using stopwords improved scoring performance on 4 out 5 measurements except on cos.w4 correlation (as in Table 29).

Since using stopwords improved performance in BOW, it was natural to wonder whether using stopwords in Wn1st can improve performance because the original Wn1st setup did not remove stopwords. Therefore the author experimented with Wn1st(stop) in which stopwords were first removed from text and then remaining words were matched to synstes. This option does not apply to Wnpos because we need the full sentence to run POS tagger. The experiment, Wn1st(stop), showed that it had similar performance to Wn1st, with very small performance improvements on max.cos correlation, accurary, and kappa.

The above experiments show that merging dimensions on this data set not only made Wn1st perform poorly, but also made BOW(stemming,non-stop) have poor performance. It again proved that merging dimensions can decrease performance. Using stoplist on BOW can improve performance but it did not cause much performance change for Wn1st. These experiments further help people understand the effects of using stoplist and merging dimensions on final performance.

The author continued analyzing the poor performance of Wn1st by looking into a case of synonyms. Synonyms share similar meanings but they may reflect different language levels. For example, “*begin*” and “*commence*” are synonyms, while the former is an everyday word and the latter one usually indicates a higher proficiency of English vocabulary. The two words are two different dimensions in BOW representation whereas they are merged to one synset dimension in Wnpos, which means they are assigned the same weight regardless of their original word format.

The author illustrates this synonym issue by using transcript 7589166-VB531098, a score 4 response. It was scored correctly to 4 by BOW and incorrectly scored to 3 by Wn1st. This is a snippet:

"University has increased the fees by eight percent. Several reasons have been ***cited*** *{mention.v.01} for that."*

Word "*cited*" was mapped to the *{mention.v.01}* synset that also subsumes word "*mention*". In fact, "*mention*" is used widely in other transcripts while"*cited*" is only used in this transcript. Though "*cited*" may not be a much more sophisticated word than "*mention*", it still suggests this speaker possesses a good knowledge of English since this is a relatively uncommon word in the corpus, while still of the same meaning to the common word "*mention*". Therefore it is expected that "*cited*" obtains a high weight in the document. In the BOW representation, weight of "*cited*" was 0.1653 due to its high idf in the corpus (a rare word); in Wn1st, weight of synset *{mention.v.01}* was low as 0.1090, because after merging synonyms, one synonym can occur in more transcripts, which caused lower idf than in BOW. Thus in the Wn1st representation, although the representation unit was more meaningful, the weights of synonyms were less helpful to content scoring than BOW, because some useful words that were indicative of speaking proficiency lose their high weight when merged with other less useful ones. The author considers that the incorrect scoring from Wn1st was partly due to the lowered weight of word "*cited*" by this representation.

4.4.4 In-Depth Analysis on OntoReason

As discussed before, the OntoReason-WordNet approach had lower performance compared to the BOW and ONTO-WordNet approach except on the

cos.w4 correlation measurement. The same thing happened to Wikipedia: ONTO-Wiki outperformed OntoReason-Wikipedia except on cos.w4 correlation. The author proposed that the characteristics of the scoring model, e-rater model, may be the reason causing poor performance of OntoReason.

The e-rater scoring model takes outputs of the representation process, which are vectors of test documents and score level training documents. The score level vectors are established for each score level group, with the assumption that these vectors are representative vector for the corresponding score levels. In fact, score level 4 vector is representative for the prompt topic whereas other score level vectors may not be. Given a set of high-scored transcripts, the useful terms for the score level can be extracted from them; however, given a pool of low-scored transcripts, the terms extracted from them is actually not representative for the score level. It is actually difficult to establish a representative vector for low-scored speech, because terms used in low-scored essays vary greatly. A set of relevant terms to the topic can be summarized from high-scored transcripts, but as there are various ways of composing low-quality speech such as using different sets of irrelevant terms, it is extremely difficult, if not impossible, to construct a representative vector for low-scored transcripts. In other words, the score level vectors generated from high-scored transcripts is representative but the vectors extracted from low-scored transcripts are not representative for their score levels.

For example, score level 2 vector is generated from responses scored 2, whose terms are usually irrelevant to the prompt. Because they are often short in length, the vocabulary size of score level 2 responses is often small as well. Then given a low-scored test file, which also contains irrelevant terms to the prompts but different from

the terms in score level 2 responses, it is represented as vector. Then cosine similarity is computed between the test vector and the three score level vectors for score predicting. The similarity between the test vector and score level 2 vector will be fairly small because they do not share much vocabulary. Additionally, due to the small data set and short length of score level 2 transcripts, the vector size of score level 2 is often small as well. Therefore the cosine similarity between the test vector and score level 2 vector will be small. On the contrary, the score level 3 and 4 vectors have a larger vector size because they have more samples in the data set and they are generally much longer than the score 2 transcripts. The larger vector size will lead to more vocabulary match with the test vector than score 2 vector, and thus will result in larger similarity between the test vector and the score level 3 and 4 vectors than the score 2 vector. Then the predicted score of this test file is probably 3 or 4 because their score level vectors have higher cosine similarity with the test vector than score 2. However, actually the test transcript contains irrelevant content and should not be scored this high, and a scoring error occurs due to the inaccurate representation of low-scored transcripts.

Furthermore, the reasoning approach can worsen the situation by introducing more vector dimensions in the score level vectors. As the vector of low-scored responses is not representative for the score level, it is even more harmful to the representation to expand the vectors by adding concepts that are similar to the concepts in the responses but are actually not relevant to the topic. In other words, the inaccurate representation of low-scored transcripts is propagated and enlarged in the unknown term reasoning process, resulting in even more inaccurate representation.

Therefore the performance of the OntoReason approach decreased compared to the ones that do not use reasoning methods (BOW and ONTO). These interpretations are especially helpful for understanding the poor performance of OntoReason-WordNet; for OntoReason-Wikipedia, the ONTO-Wiki did not return good vector representation anyway, so its reasoning performance was even worse.

4.4.5 Beyond Averaged Results

BOW outperformed ontology-based approaches at many times, however, there were some circumstances that ontology-based approaches exceeded BOW on some measurement and at some score level. The hypotheses were answered based upon averaged performance, while here we look into the granular results and find situations when ontology-based approaches performed well.

First, performance of Wnpos looked inferior to the baseline BOW(tfidf). However, Wnpos was superior to BOW(tfidf) on classifying score 3 speech, as shown by F measure, precision, and recall values (Table 30). Precision and recall are also machine learning evaluation metrics and are used to compute F measure here.

Table 30. Performance on score level 4.

	F measure (score=3)	Precision (score=3)	Recall (score=3)
BOW(tfidf)	0.5158	0.505	0.529
Wn1st	0.5131	0.514	0.5124
Wnpos	**0.5396**	**0.5191**	**0.5625**

Second, WnReasoning(Wn1st, Lin) and WnReasoning(Wn1st, Dft) had better performance than BOW(tfidf) on identifying score 2 speech.

Table 31. Performance on identifying score 2 speech.

	F measure (score=2)	Precision (score=2)	Recall (score=2)
BOW(tfidf)	0.4281	**0.4626**	0.3525
WNreasoning(Wn1st, Lin)	0.4563	0.3972	0.54
WNreasoning(Wn1st, Dft)	**0.4634**	0.4003	**0.5525**

Third, WikiReasoning(Content) achieved higher recall than BOW(tfidf) on score level 2 classification: 0.4565 for WikiReasoning(Content) and 0.374 for BOW(tfidf).

Fourth, Combined(Wn1st,BOW) improved over BOW(tfidf) on classifying score levels 2 and 3 transcripts. It indicates that ONTO-WordNet approaches produced better results over BOW(tfidf) on score 2 or score 3 classification, but not on score 4 classification.

Table 32. Performance on identifying score 2 and 3 transcripts.

	F measure (score=2)	Precision (score=2)	Recall (score=2)
BOW(tfidf)	0.4281	0.4626	0.3525
Combined(Wn1st, BOW)	**0.4454**	**0.5322**	**0.385**
	F measure (score=3)	Precision (score=3)	Recall (score=3)
BOW(tfidf)	0.5158	0.505	0.529
Combined(Wn1st, BOW)	**0.5286**	**0.5084**	**0.5511**

The author would like to point out though non-BOW approaches outperform BOW in some circumstances, namely on a specific measurement at a specific score level, the averaged performance was still the main evidence used to determine quality of representation approaches.

4.4.6 Analysis of Selected Cases

The author aggregated predicted scores resultant from different approaches along with actual scores to analyze patterns in different outputs from the same test transcript. Then the author identified some important transcript samples that have large performance variations on different representation approaches. For example, those that were predicted incorrectly under BOW(tfidf) representation but were predicted correctly under Wn1st representation, to analyze and understand reasons why one representation approach outperformed another approach on a specific transcript.

Case I (Wn1st and Wnpos correct, while BOW(tfidf) incorrect)

Transcript 7545576-VB531098, whose actual score is 3, was incorrectly scored to 2 by BOW(tfidf) and correctly scored to 3 by Wn1st. The transcript content is:

> *Uh the woman think that it is a good **idea** uh to make uh the **tuition** higher uh because she think that uh it will **allow** to the university uh to make a better condition for studying. Uh for example to make uh **groups** of **student smaller** and so each student uh will **get** uh personal **attention** that it is very important for him. Uh also uh she **complains** that the **equipment** uh is not new enough uh **out-of-date** and so it doesn't allow her uh to be **prepared** enough for her **job** for her **job after** graduation. Uh he think that the **facilities** are limited uh in the university and the higher **tuition** uh*

Using the Wn1st approach, 45 synsets were extracted to represent this transcript. Among the 45 synsets, 15 of them were merged synsets, by which the author refers to synsets subsuming more than one word in the corpus. Merged synsets (bold words in the above text) mean synonyms were combined to one dimension so that the power of concept-based representation can be released. For example, "*allow*" in the transcript shared a synset with "*let*" in other transcripts, "*get*" in the transcript was matched to the same synset as "*got*", "*get*", and "*acquire*" in other transcripts.

The author conjectures that this transcript was correctly scored by Wn1st because a large portion of matched synsets were merged synsets (25/45=55.56%). However, more evidence is needed to judge this statement.

This same transcript was also correctly scored to 3 when using the Wnpos approach. 22 synsets were found to be merged synsets, which was also a large percentage of the total (22/36=61.11%). The merged synsets were similar to the ones in Wn1st, such as "allow" and "get" share synsets with their synonymous words in other transcripts, except that some synsets were excluded from the vector because Wnpos employs POS filtering.

Case II (ONTOReason-WordNet methods correct, while BOW(tfidf) and Wn1st wrong)

The author chose a transcript that was incorrectly scored by BOW(tfidf) and Wn1st but correctly scored by WNreasoning(Wn1st, Path), WNreasoning(Wn1st, Lin), and WNreasoning(Wn1st, Dft). Out of the 6 transcripts of prompt 101 that met this requirement, she selected transcript 7567976-VB531101 for analysis. The actual score is 4, while BOW(tfidf) and Wn1st graded it to 3, and the 3 reasoning methods correctly assigned the grade to 4. The transcript content is (with word strings of unknown concepts in bold):

> *The automobiles and radio uh has contributed to the common culture in the U S A um as **follow**. Uh for example the automobiles ava- become available in nineteen twenties and um uh uh people start to travel more and uh they **taking** vacation to another part of the country. So um they start to change attitudes whe- when they meet people from different cultures and they start to adopted uh behavior from uh big cities people. Um the same thing the radios people start to shared um uh the experience by uh listen to the same artists and uh the same popular radio programs, the same news that are reported uh **daily** and um which are better than the newspaper.*

Given the Wn1st vector of the transcript, 3 synset were found to be unknown concepts to the score level 4 vector of the prompt. These unknown concepts were

{daily.r.01}, {taking.n.01}, {follow.v.01}

Synset *{daily.r.01}* was ignored in reasoning because the structure of adverbs and adjectives follows a cluster fashion instead of strict hierarchy. Weights of other two synsets can be guessed by using their 5 most similar concepts in the score 4 vector. Table 33 lists both unknown synsets and their similar synsets under different similarity options.

Table 33. Unknown synsets and their most similar synsets in the score 4 vector.

Similarity Option	**Unknown synset**	**Most similar synsets in the score 4 vector of prompt 101**
Path	{*taking.n.01*}	{*production.n.01*}, {*discovery.n.01*}, {*event.n.01*}, {*communication.n.01*}, {*acquiring.n.01*}
	{*follow.v.01*}	{*travel.v.01*}, {*come.v.01*}, {*be.v.01*}, {*know.v.01*}, {*necessitate.v.01*}
Lin	{*taking.n.01*}	{*woman.v.01*}, {*World_Health_Organization.n.01*}, {*dressing.n.01*}, {*parlance.n.01*}, {*component.n.01*}
	{*follow.v.01*}	{*travel.v.01*}, {*come.v.01*}, {*associate.v.01*}, {*imitate.v.01*}, {*explain.v.01*}

The default similarity option is not included in Table 33 because that method simply averaged weights of all synsets in the score 4 vector. We can tell that for Path similarity, synsets similar to *{taking.n.01}* and *{follow.v.01}* are quite relevant to the unknown concepts; for Lin similarity, while the similar synsets for *{follow.v.01}* are semantically close, similar synsets for *{taking.n.01}* are not that relevant, especially compared to similar synsets in the Path similarity option.

The OntoReason-WordNet methods located the correct score level vector for the test transcript and thus achieved better performance than BOW(tfidf) and Wn1st on this transcript. These reasoning methods succeeded because of the high relevance of the unknown concepts to the prompt theme. That is to say, with the theme of prompt 101 being the contribution of the automobile and radio to unification in US, the unknown synsets, {*taking.n.01*} and {*follow.v.01*}, were relevant to the topic. Wn1st and BOW(tfidf) discarded these words and synsets in representation, whereas these unknown concepts were assigned weights in the reasoning approaches; since they were topic relevant, the expanded vector were raised closer to the score level 4 vector.

Therefore, the author argues the reasoning methods can increase performance when meeting this condition:

The unknown concepts are topically relevant.

This can effectively add more relevant concepts to the vector for more meaningful representation. If unknown concepts are not so topically relevant, guessing a non-topic weight can bring adverse effect by introducing noise because reasoning may overestimate their low weight.

Case III (BOW(tfidf) correct, while Wn1st, Wnpos and Wiki incorrect)

The author selected one transcript whose scores were incorrectly predicted by some ONTO approaches but predicted correct by BOW(tfidf). The transcript, 7667185-VB531098, was incorrectly scored as 4 by Wn1st, Wnpos, and DirectWiki, and was correctly scored as 2 by BOW(tfidf). Its original content is:

> *The woman uh opinion is to be in favor of uh uh increasing the uh tuition fee because she is uh afraid she couldn't find a job uh when she graduate because she is working in the laboratory with the out-of-date equipment about a microbiology.*
> *So if she is uh have a competition with other uh uh people apply for the job, uh probably the other is already graduate from a new uh uh facility, a new laboratory about microbiology uh and she is uh afraid about that. So uh she is in favor uh uh of uh increasing this tuition fee and she think that increasing will be uh good for the university.*

Since the positives and negatives of Wn1st and Wnpos have been extensively discussed in previous sections, the author focuses on investigating DirectWiki result here. DirectWiki only found 2 Wikipedia concepts within the text, scoring the transcript as 4:

> *The woman uh opinion is to be in favor of uh uh increasing the uh tuition fee because she is uh afraid she couldn't find a job uh when she graduate because she is working in the laboratory with the out-of-date equipment about a microbiology . So if she is uh have a competition with other uh uh people apply for the job, uh probably the other is already graduate from a new uh uh facility, a new laboratory about microbiology uh and she is uh afraid about that. So uh she is in favor uh uh of uh increasing this tuition fee and she think that increasing will be uh good for the university.*

The Wiki concepts identified from the transcript were "tuition fee" and "microbiology"; it is not surprising that there were only two because Wikipedia is a

collaborative encyclopedia, not a lexicon or dictionary. It seeks to cover all human knowledge and makes each topic a Wiki article ("Wikipedia," 2013). The majority of Wiki concepts are topical nouns and therefore other terms cannot be matched in Wikipedia. As discussed before, the errors by the Wikifier package also contributed to the small number of returned concepts. For example, concepts "*woman*" and "*laboratory*" were included as article titles in Wikipedia but they were not identified by the Wikifier toolkit. The author manually searched for words and phrases occurring in the transcript in the Wikipedia web interface, and found these matches highlighted below, which reflects a large number of concepts missed by Wikifier:

> *The woman uh opinion is to be in favor of uh uh increasing the uh tuition fee because she is uh afraid she couldn't find a job uh when she graduate because she is working in the laboratory with the out-of-date equipment about a microbiology . So if she is uh have a competition with other uh uh people apply for the job, uh probably the other is already graduate from a new uh uh facility, a new laboratory about microbiology uh and she is uh afraid about that. So uh she is in favor uh uh of uh increasing this tuition fee and she think that increasing will be uh good for the university.*

Case IV (an outlier file)

Transcript 7549468-VB531098 is an outlier, scored 3 by the human grader but is actually a file of nonsense. It only contains a textual comment,

"*score this response as a three*",

probably because of mistaken operations by the human grader when copying and pasting text. Though the content is not the original response, this can serve as a good example of an irrelevant transcript that should be scored as 2.

In the experiments, this transcript was scored to 2 in BOW(tfidf), Wn1st, Wnpos, WNreasoning(Wn1st, Path) and so on, while it was incorrectly graded by approaches such as WNreasoning(Wn1st, Dft).

4.4.7 Statistical Significance Test

The author primarily compared averaged performance when comparing two representation approaches. She did not employ statistical significance test as basis for comparison because the small sample size (4 prompts in total) did not provide sufficient evidence leading to firm conclusions. Due to this reason, instead of running significance tests for all the comparisons, the author only ran one t-test to compare an ontology-based representation and BOW baseline.

When running significance test on two approaches, we need to run a paired sample t test, with each prompt as a sample. There will be one t-test for each measurement aspect, and thus 5 t-tests in total, when comparing two approaches using t-tests.

The author ran a t-test to examine whether the combined (ontology-based) approach Combined(BOW=0.7,Wn1st=0.3) significantly improved performance over the BOW(tfidf) baseline, with awareness that the small sample size may make the t-test result less informative. Therefore, for each measurement, a paired sample t-test was conducted between Combined (BOW=0.7,Wn1st=0.3) and BOW(tfidf), and the results are shown in Table 34. The t-test results were not conclusive:

On max.cos correlation, there was not a significant difference in the scores for Combined(BOW=0.7,Wn1st=0.3) (Mean=0.3704, SD=0.0394) and BOW(tfidf) (Mean=0.3494, SD=0.0437) approaches; t(6)= 0.7142, p= 0.5019.

On cos.w4 correlation, there was not significant difference in the scores for Combined(BOW=0.7,Wn1st=0.3) (Mean=0.3651, SD=0.0455) and BOW(tfidf) (Mean=0.3556, SD=0.0515) approaches; t(6)= 0.2759, p= 0.7918.

On F measure, there was not significant difference in the scores for Combined(BOW=0.7,Wn1st=0.3) (Mean=0.4710, SD=0.0236) and BOW(tfidf) (Mean=0.4627, SD=0.0114) approaches; t(6)= 0.6337, p=0.5497 .

On accuracy, there was not significant difference in the scores for Combined(BOW=0.7,Wn1st=0.3) (Mean=0.4595, SD=0.0355) and BOW(tfidf) (Mean=0.4786, SD=0.0205) approaches; t(6)= 0.7820, p= 0.4639.

On kappa, there was not significant difference in the scores for Combined(BOW=0.7,Wn1st=0.3) (Mean=0.4870, SD=0.0286) and BOW(tfidf) (Mean=0.3441, SD=0.0448) approaches; t(6)= 0.4767 , p=0.6505.

The above result report shows that no significant difference was found on any measurement aspect. The author was not able to conclude since the small sample size can be a contributing factor to the result, and it was difficult to tell whether Combined(BOW=0.7,Wn1st=0.3) made a significant difference from BOW(tfidf) based on the current evidence.

Table 34. Significance test results.

Significance test	
BOW(tfidf) vs. Combined(BOW=0.7,Wn1st=0.3)	
On max.cos corr.	Mean(BOW)=0.3494, SD(BOW)=0.0437 Mean(Combined)= 0.3704, SD(Combined)= 0.0394
	t(6)= 0.7142, p= 0.5019 (two-tailed), 95% confidence interval: from -0.0929 to 0.0509
On cos.w4 corr.	Mean(BOW)=0.3556, SD(BOW)=0.0515 Mean(Combined)= 0.3651, SD(Combined)= 0.0455
	t(6)= 0.2759, p= 0.7918 (two-tailed), 95% confidence interval: from -0.0935 to 0.0745
On F measure	Mean(BOW)=0.4627, SD(BOW)=0.0114 Mean(Combined)= 0.4710, SD(Combined)= 0.0236
	t(6)= 0.6337, p= 0.5497 (two-tailed), 95% confidence interval: from -0.0404 to 0.0238
On accuracy	Mean(BOW)=0.4786, SD(BOW)=0.0205 Mean(Combined)=0.4595, SD(Combined)= 0.0355
	t(6)=0.7820, p=0.4639 (two-tailed), 95% confidence interval: from -0.0311 to 0.0692
On kappa	Mean(BOW)=0.3441, SD(BOW)=0.0448 Mean(Combined)= 0.4870, SD(Combined)= 0.0286
	t(6)= 0.4767, p= 0.6505 (two-tailed), 95% confidence interval: from -0.0514 to 0.0346

4.4.8 Prompt-specific Analysis

The averaged performance over all 4 prompts compared representation approaches by using averages. Besides observing average, observing and comparing results on prompt level can also provide insights to the study. The author thus looked at performance values of each prompt and examined whether there were consistent trends on prompt level.

The table below aggregates prompt-specific results on each measurement from several important representation approaches.

Table 35. Performance on each individual prompt.

Experiment	Prompt	max.cos corr.	cos.w4 corr.	F measure	accuracy	kappa
BOW(tfidf)	098	0.3728	0.4293	0.4548	0.4717	0.3669
	099	0.3194	0.3423	0.4511	0.4608	0.3118
	100	0.3065	0.3094	0.471	0.5081	0.3019
	101	0.3988	0.3414	0.4739	0.4737	0.3959
ONTO-WordNet (Wn1st)	098	0.2462	0.3857	0.3971	0.4119	0.2441
	099	0.2619	0.3504	0.4625	0.4804	0.2594
	100	0.2514	0.3142	0.4604	0.4919	0.246
	101	0.3149	0.3409	0.4488	0.4539	0.3129
ONTO-Wiki (DirectWiki)	098	0.1187	0.2156	0.3733	0.3994	0.1162
	099	0.2186	0.2448	0.3623	0.3693	0.2173
	100	0.1456	0.1977	0.3327	0.3528	0.1239
	101	0.0948	0.0901	0.3274	0.3783	0.0888
OntoReason-WordNet (WNreasoning(Wn1st, Path))	098	0.2553	0.4088	0.4223	0.4214	0.2522
	099	0.2325	0.3928	0.4098	0.4118	0.2286
	100	0.2102	0.3318	0.4566	0.4693	0.2093
	101	0.3065	0.3547	0.4482	0.4474	0.3043
OntoReason-Wikipedia (WikiReasoning(Content))	098	0.0815	0.2286	0.3382	0.3428	0.0797
	099	0.2647	0.2795	0.364	0.3595	0.2616
	100	0.1402	0.2092	0.2989	0.3301	0.1078
	101	0.0006	0.0544	0.3333	0.3553	0.0006
OntoWordNet (Combined (BOW=0.7, Wn1st=0.3))	098	0.3593	0.4299	0.4357	0.4497	0.3511
	099	0.3919	0.356	0.4843	0.4967	0.3868
	100	0.3204	0.3237	0.4831	0.5178	0.3145
	101	0.4099	0.3507	0.4809	0.4836	0.4067

The author compared approaches by examining the 5 performance measures at prompt-level, and summarizes these findings:

Wn1st vs. BOW(tfidf). Wn1st only used WordNet synset vectors, and the prompt-level results showed that for each prompt, Wn1st still had lower performance than BOW, except in some sporadic cases, e.g. F measure for prompt 099, where Wn1st had higher values.

DirectWiki vs. BOW(tfidf). Down to the prompt level, DirectWiki still had inferior performance on all the measurements for each prompt.

WNreasoning(Wn1st, Path) vs. Wn1st. The comparison showed that for prompt 098, WNreasoning(Wn1st, Path) exceeded performance of Wn1st on all measurements.

For the other 3 prompts, Wn1st had better performance over WNreasoning(Wn1st, Path) on all measurements but cos.w4 correlation.

WikiReasoning(Content) vs. DirectWiki. No consistent trend was detected on the prompt level for these two approaches. It is worth mentioning that for prompt 101 WikiReasoning(Content) had lower cos.w4 correlation than DirectWiki, whereas for other prompts WikiReasoning(Content) had higher cos.w4 correlation values.

Combined (BOW=0.7, Wn1st=0.3) vs. BOW(tfidf). Except for prompt 098, for all the other 3 prompts, Combined (BOW=0.7, Wn1st=0.3) had higher values than BOW(tfidf). For prompt 098, Combined (BOW=0.7, Wn1st=0.3), Combined (BOW=0.7, Wn1st=0.3) had slightly higher (nearly the same) cos.w4 correlation than BOW(tfidf), and on the other 4 measurements, BOW(tfidf) exceeded Combined (BOW=0.7, Wn1st=0.3).

4.5 Naïve Bayes (NB) Scoring Model

In addition to the e-rater model, the author also applied the NB model on the word and synset vectors to examine whether NB model improves scoring performance. The author used the NaiveBayesMultimonial model, a model for multi-nomial class, in the Weka toolkit for this task. It turns out that on each prompt, the NB model classified all instances to score 3 for BOW and classifies most instances to score 3 for Wn1st. The NB classifiers favored strongly towards the score 3 class on this data set. Table 36 and Table 37 show the confusion matrices for prompt 099 on the two representations.

Table 36. Confusion matrix for prompt 099 (representation=BOW, machine learning=NB)

	Score=2 (predicted)	Score=3 (predicted)	Score=4 (predicted)	Sum
Score=2 (actual)	0	93	0	93
Score=3 (actual)	0	144	0	144
Score=4 (actual)	0	69	0	69
Sum	0	306	0	306

Table 37. Confusion matrix for prompt 099 (representation=Wn1st, machine learning=NB)

	Score=2 (predicted)	Score=3 (predicted)	Score=4 (predicted)	Sum
Score=2 (actual)	2	91	0	93
Score=3 (actual)	2	142	0	144
Score=4 (actual)	0	69	0	69
Sum	4	302	0	306

NB had poor classification performance because the document-weight matrix was sparse, especially for score levels 2 and 4, the probability of some words occurring in these 2 categories were very low. For example, transcript 7600879-VB531099 has the word "*trying*", which only occurs in score 3 vector but not score 2 and 4 vectors. The NB algorithm assigned a low probability (smoothing) to this word in score 2 and 4 vector to avoid having a 0 when multiplying probabilities. The probabilities of this word in scores 2, 3, 4 were:

$7.9041*10^{-4}$ (in score 2), 0.0011 (in score 3), $8.1010*10^{-4}$ (in score 4)

Due to the nature of the corpus, there are more score 3 transcripts than score 2 and 4 transcripts, and therefore occasionally some words from a test transcript only occur in score 3 vector but not the other two levels. The non-occurrence of these words at these score levels may cause the result that a test transcript has a higher probability of belonging to the score 3 class. The same thing happened to the Wn1st representation.

The author lists the performance measurements below, which are poor and unbalanced results. Cos.w4 is not included because it is a correlation measurement specifically designed for the e-rater model.

Table 38. NB model performance.

	Avg. max.cos corr.	Avg. F measure corr.	Avg. accuracy	Avg. kappa
BOW (NB model)	n/a	0.2178	0.4800	0.0028
Wn1st (NB model)	n/a	0.2212	0.4800	0.0081

4.6 Summary

The author aggregated the performances from the best parameter option of each representation approach in Table 39. As we can see, each representation came with a set of parameters and thus representation performances were tied to their parameter setup. That is to say, to compare two approaches, it is important to understand what the parameters are and what they mean.

As the author measures performance from several different aspects, the measurements on the one hand inspects results from multiple lenses, but on the other hand they complicated the performance comparison because of the volume of the results. It made it more difficult to judge which approach outperformed another. In the case that a representation has better performance than another one on all measurements, we may say that it is better than the other one (e.g. BOW(tfidf) > LSA(k=40)). In some other cases, where one representation performed better than another one on some measurements but not all of them, it was hard to determine whether it was a better approach (e.g. BOW(tfidf) vs. Combined(Wn1st, BOW)).

Table 39. Performance from best parameter option of each representation approach.

	Experiment	**Avg. max.cos corr.**	**Avg. cos.w4 corr.**	**Avg. F measure**	**Avg. accuracy**	**Avg. kappa**
BOW	BOW(tfidf)	0.3494	0.3556	0.4627	0.4786	0.3441
LSA	LSA(k=40)	0.2506	0.151	0.3931	0.4442	0.2393
ONTO-WordNet	Combined (Wn1st, BOW)	0.343	0.3653	0.4631	0.4815	0.3382
Onto-Wiki	DirectWiki	0.1444	0.187	0.3489	0.3749	0.1366
OntoReason-WordNet	WnReasoning (Wn1st, Path)	0.2422	**0.3864**	0.4246	0.4249	0.2381
OntoReason-Wikipedia	WikiReasoning (Content)	0.1217	0.1929	0.3336	0.3469	0.1124
Combined	Combined (BOW=0.7, Wn1st=0.3)	**0.3704**	0.3651	**0.4710**	**0.4869**	**0.3648**

A number of points were discussed in the previous sections, and the author summarizes some of them to conclude the chapter here:

First, the line charts showed that max.cos correlation and kappa exhibited similar trends, and F measure and accuracy also had similar curves (e.g. Figure 12 and Figure 13). This is possibly because max.cos correlation and kappa both measure ordinal classification performance, whereas F measure and accuracy are general classification measures.

Second, cos.w4 correlation, max.cos correlation, and kappa were sensitive evaluation measurements. They were susceptible to representation and parameter setup while the other two, F measure and accuracy, did not change that much. For example, in Figure 11 and Figure 12, cos.w4 correlation, max.cos correlation, and kappa had a “rockier” line than the other two. This is probably because these three measurements contain ordinal information and thus reflect more performance changes.

Third, reasoning methods (OntoReason) made cos.w4 correlation a better indicator of speech scores. Though other performance measurements decreased in OntoReason, but cos.w4 correlation of OntoReason increased over ONTO, in both WordNet and Wikipedia cases. Since cos.w4 correlation measures distance between a test transcript and score 4 transcripts, the results may indicate that cos.w4 correlation improves the accuracy of distance measurement with score 4 vector.

Fourth, the scoring models built from the vector representations tended to classify instances to score 3. Table 7 displays a confusion matrix from BOW representation, in which a large number of the instances were classified to score 3, and the same situation happened to most other experiments. The author thinks it was partly due to the relatively large number of score 3 transcripts, compared to scores 2 and 4. A larger number of files made score 3 contain more terms, and thus when measuring with cosine similarity, score 3 vector had more chance of being close to a test transcript than other score levels because it had a larger vector size.

Fifth, though ontology-based representation approaches are theoretically sound, they did not outperform the baselines all the time in practice. It showed that BOW was simple but robust, while ONTO and OntoReason approaches can improve over BOW under some circumstances, including:

- ✓ Combining WordNet and BOW vectors with a weighting strategy by using multipliers on different vector similarities
- ✓ Improving cos.w4 correlation performance by using ONTO-Reason

For the best combined approach, Combined (BOW=0.7, Wn1st=0.3), its relative improvement over BOW was 6.01% on max.cos correlation, 2.67% on cos.w4 correlation, 1.79% on F measure, 1.73% on accuracy, and 6.02% on kappa. For the

best cos.w4 correlation in OntoReason, it improved relatively 4.61% over the cos.w4 correlation in BOW.

Sixth, combining WordNet and BOW vectors increased performance over only using WordNet but not necessarily over BOW; combining Wikipedia and BOW vectors improved over only using Wikipedia but not over BOW. Combining the three, BOW + WordNet + Wikipedia, made performance lower than BOW. It indicates that currently Wikipedia representation brings adverse effect to scoring performance; while combining BOW + WordNet in a good proportion can enhance performance over BOW.

Lastly, though the responses to the hypotheses have addressed the research questions proposed in Chapter 1, the author briefly summarizes them here.

RQs:

Does ontology help?

how does ONTO affect speech scoring?

How does OntoReason affect speech scoring?

The answer is that it helps when combined with BOW approach. It does not help when only using concept vectors for representation. It seems the combined representation is a better one, which is also theoretically sound because it is a richer representation than either concept-only or word-only representations. Practically, combined representation expands the vector and strengthens important dimensions. On the other hand, when only using concepts in the vector, the benefits brought by concepts are counteracted by problems brought in and, thus, ontology-based representation results in a less competitive method than the BOW approach. Though not a huge improvement, combining different ontology and word vectors can increase

performance over the BOW approach, based on the result of averaged performance. OntoReason generally decreases scoring performance, but it performs well on cos.w4 correlation and makes cos.w4 a better content feature in the speech construct.

5. DISCUSSION AND CONCLUSION

The author has illustrated how ontologies help text representation from a theoretical perspective in Chapters 2 and 3, and reported the experiment in Chapter 4, which empirically evaluated to what extent ontologies might help speech scoring in the context of this study. This chapter summarizes the empirical findings and discusses their theoretical implications with future studies.

5.1 The Role of Ontologies in Text Classification

Content scoring of speech is essentially a text classification task aided by natural language processing. It goes through the procedures that first represent content in a computable format and then apply classification algorithms to assign scores based on the representation. Text representation is the middle product in this workflow. The vectors resulted from text representation are used as input of the classification module: text => representation => classification. The supervised text classification algorithms was used (also called supervised “machine learning”) in this study for the classification procedure.

Theoretically, text representation approaches determine representation units and weights, which are also important factors in text classification. The experimental results in chapter 4 have demonstrated that classification results vary due to the change in representation approaches. It is also noteworthy that speech scoring is the result of two consecutive modules: representation and classification. These findings suggest that performance differences between representation approaches could have been affected by both modules. The author controls the effect of machine learning algorithms by fixing the machine learning algorithm when representing the same content using different

approaches. Since machine learning is necessary in the system, this is the best that the author can do to control its effect.

In the context of speech scoring, the above discussion implies that better scoring performance *may* be an indicator of a better representation approach. We need to be cautious about this, however, because there is a layer of machine learning that can affect scoring performance. With this in mind, the author will discuss the role of ontologies in text processing based on the experiment results.

As reflected by the experimental results in chapter 4, the ontology-based representations, including ONTO and OntoReason, showed slight improvement over the baselines in some circumstances. For example, some ONTO-WordNet parameter options such as combining different types of vectors had higher performance on average than those from BOW baseline (in Table 25). However, under many parameter options, the ontology-based approaches actually performed less well than the BOW baseline.

Previous studies that employed concept- or phrase-based representations on text related tasks had produced similarly mixed results. Example representation units included phrases, clusters of phrases, WordNet synsets, or Wikipedia titles (Lewis, 1990; Scott & Matwin, 1999; Hotho et al., 2003a; Zhang, 2009; Gabrilovich, 2007). These studies show that using synsets only or phrases in vectors does not seem to consistently improve text classification performance over the bag-of-words baselines, but combining different vectors or classifiers can improve it. Past research also had similar findings for using phrases and concepts from ontologies. Examples include a combination of words, phrases, and clusters of phrases (Lewis, 1990), a combination of

WordNet synset and word vectors (Hotho et al., 2003a), a voting scheme from classifiers from synset and word vectors (Scott and Matwin, 1999), and words combined with Wikipedia concepts or categories (Zhang, 2009; Gabrilovich, 2007). The fact that these combined representations achieved the best performance indicate that combining multiple representation approaches is a relatively robust means to enhance representation quality and text classification performance.

This study validates that the combined vectors have a positive effect, among which the combined WordNet and word vectors stands out as the best performance. From the above discussion and the experimental results, it is reasonable to conclude that ontologies are useful in enhancing text related tasks such as text classification when combined with the bag-of-words approach. Ontologies are not supposed to substitute bag-of-words, but to strengthen important concepts and extend word vectors for a richer representation instead. Adding ontology-based vectors can further augment weights of important concepts in text and thus results in a better representation than using words or concepts alone.

It should be pointed out that the speech scoring task in this study is a special type of text classification. Text classification typically assigns a predefined label to an unseen document (Scott & Matwin, 1990). In speech scoring, the predefined labels involves several ordinal score levels. Text classification algorithms usually handle the nominal class, in which class labels are at the same level, that is, no ranking between classes. However, ordinal classes contain information about the rank of class labels. For example, in this study, the “score 4” class was ranked higher than the “score 3” class since “score 4” means higher speech quality. This study employed a regular

machine learning algorithm (e-rater, similar to the Rocchio algorithm) without considering the ordinal property of class labels, which means that each score level is treated equally. This implies that the effect of ontology-based representation on text calssification is subject to speech scoring under a nominal classifier rather than an ordinal classifier.

5.2 The Role of Ontology in Text Representation

Text is a string of words on the surface and can be easily understood by human beings. Such a string of words, however, can be difficult for machines to correctly and accurately recognize the underlying semantics because text semantics that is explicit to human cognitive ability can be implicit for computers to process. The use of ontologies can help converting implicit semantics in text into explicit semantics. This was achieved through the vectors generated from different representation approaches, which extracts linguistic units (i.e. words, latent concepts, and explicit concepts from ontologies) and arranges them in vectors. Since ontologies are representations of domain knowledge in formalized languages and contain explicit semantics that typically includes concepts and relations, text representations can benefit from using them to reveal semantics in text, for example, extracting concepts in this study.

Ontology-based representations enhance the meaningfulness of the vector dimensions over traditional bag-of-words since concepts are richer representation units than words. Besides meaningfulness, the use of concept vectors also results in dimensionality reduction in this study. As in Table 20, Wn1st vector has fewer dimensions on score level 4 compared with the BOW vectors. However, in Wikipedia vectors, the dimensionality reduction is also partly due to the imprecise match from text

to Wikipedia concepts. As illustrated in case III in section 4.4.6, only 2 Wikipedia concepts were identified in that transcript whereas 45 WordNet synsets were extracted from the same transcript. From the perspective of representation, too few dimensions indicate a weak representation of the content and may cause further performance drop in speech scoring. The low performance of DirectWiki (using Wikipedia concept vectors) is likely to be the result of too few dimensions in representation.

In this study, ontology-based representation along with other representations was in vector style. This means that concepts are considered orthogonal to each other, but in reality concepts are actually connected to each other via some path in an ontology and these connections have not been completely reflected in the representation. Although we have groups of synonyms (e.g. WordNet synsets), the rich semantics embedded in the ontologies was exploited only in a limited way so far. For example, WordNet noun synsets have hypernym, hyponym, and sister synsets, which are potentially useful information in computing semantic similarity between documents. Wikipedia contains categories and incoming and outgoing links for the Wikipedia concepts, which are also important domain knowledge for text representation. Should all these knowledge be deployed, the text representation may have been more precise and comprehensive and accomplish better performance for speech scoring or text classification.

A question remains for how we can integrate full knowledge of an ontology into text representation in a systematic way. The experience from this study suggests that the methodology of using an ontology to its full extent varies from case to case because each ontology is constructed differently and the purpose of applications also differs. So

far for text mining tasks, ontologies have been primarily used in two ways as this study did: one is to help extract concepts from text strings and the other to provide knowledge basis for computing content or concept similarity. Hotho et al.'s (2003a) study falls in the first category while also expands concept vectors by adding hypernyms of identified concepts. Lin's (1998) and Gabrilovich's (2007) work provides a new way of computing similarity scores between texts based on WordNet/Wikipedia concepts, hence are examples of the second category. Although these are the typical use of ontologies in text presentation, the knowledge in WordNet and Wikipedia can be potentially used in more ways, such as representing concepts beyond vector style and involving more ontology information in similarity calculation.

5.3 Generalization

This study investigated a specific case of text representation – speech transcript representation, but the methods used have some implications to text processing and representation research in general.

First, the methodology for content scoring of speech is a combination of representation and machine learning modules. It is a relatively standard paradigm of performing text classification, especially for the purpose of comparing representation approaches. This study focused particularly on representing content in a vector space manner, with dimensions being words, explicit concepts, or latent concepts. Vector style representations are friendly input for machine learning modules. Even if some machine learning modules are probabilistically based (such as Naïve Bayes and MaxEnt) as opposed to vector space based (e-rater, Rocchio), they can still take vector style input and perform further processing to fit the model. For example, Naïve Bayes model can

aggregate the content vectors, namely word-document matrix, to compute probability of random variables (e.g. words or concepts) in a class, which is further used to construct a Naïve Bayes classifier. This means that the representation is not only friendly input for the e-rater machine-learning model but also to other machine learning models. No matter how the machine learning models vary, these representation output is the critical input for the models, even though the vectors may be processed and used in different ways inside the models. The ontology-based representation proves to be fairly generic in the sense that they can be used as input for various types of post-processing such as machine learning.

Second, the representation methodology can potentially be applied to other text related tasks in addition to speech scoring. Due to their wide domain coverage, Wikipedia and WordNet should be able to locate concepts in text, as long as the text does not contain much uncommon terms such as jargon or highly specialized terms. From the manual inspection of the speech transcripts, the author did not find a peculiar noun or verb absent from WordNet. This may be because the content was produced during a practice test of speaking and contained descriptions for everyday life and events. The experiment also showed that, If no matching concepts can be identified for a large portion of the text, the representation quality and classification results will suffer an adverse effect.

Besides using the representation methodology for some concept-level analytics of text, the methodology can also be applied to a relevant area of speech scoring – automatic essay scoring. The baseline systems in this study follow the framework of two exemplar essay scoring systems, and to the best of the author’s knowledge, no essay

scoring system has employed ontology-based representation thus far. Essay scoring and content scoring of speech can share the same methodology, while the characteristics of essays compared with speech text may make ontology-based representation perform better on essay text.

Essay as a type of text is different from speech text in that: 1) essay text is usually longer than speech text since essay testing generally solicit longer response than in speaking test; 2) essay text contains less grammatical error because test takers have the chance to revise their writing while this is not the case in speaking tests. The first difference, longer text, makes it possible to extract more concepts from essay text and thus to generate a more representative concept vector. The second difference, less grammatical error, can lead to a concept vector with less noise and more accurate identification of concepts in text. Due to these two differences, the author thinks it is possible that performance measures such as correlation with human raters (max.cos correlation and cos.w4 correlation), can better be improved when applying ontologies in essay scoring than in speech scoring.

5.4 Contributions

This dissertation research compared different representations and parameters in the context of speech scoring. Previous automatic speech scoring research usually evaluates systems by comparing the correlation between human and automatic scoring, aiming at demonstrating that automatic systems can assign scores as accurately as human graders do. More specifically, those studies compare human-machine correlation (correlation between automatic scoring and human scoring) with human-human correlation (correlation between different human graders), and if human-

machine correlation does not exhibit a large difference from human-human correlation, then the scoring system can be claimed as a reasonably accurate system because human graders also disagree with each other to some extent.

This study differs from previous research in that it focuses on what representation approaches with which parameter setups can achieve a reasonably good performance in predicting speech scores. Due to the lack of comparison of methods in this area, the author experimented with different approaches and parameter options to learn about performance patterns. The experimental results show that BOW is a robust baseline, while ONTO and ONTO-Reason can perform well if we know when and how to use them. The most actionable implication of the study is that ontology-based representation should be combined with BOW representation to enrich content representation instead of substituting it.

Second, as mentioned in Chapter 1, the author addresses two main challenges in text representation -- meaningfulness of representation and unknown terms -- by using ontology-based approaches. These are theoretically sound representations, but need to be examined for whether they facilitate automatic speech scoring. The experiment reveals that WordNet-based representations generally outperform Wikipedia-based ontology, partly because more words can be mapped to concepts when using WordNet and thus a richer representation can be produced. Although Wikipedia-based approaches have the advantage of recognizing multi-word expressions, the small amount of identified concepts inhibited it from performing well.

Third, from the perspective of automatic speech scoring, this study was intended to contribute to content scoring of non-native spontaneous speech since scoring based

on the content of speech is less investigated than that based on acoustic features. Besides employing the two typical representation approaches from essay scoring as baselines, the author also implemented ontology-based representation on speech transcript for automatic scoring purpose, which made a unique contribution to our understanding of content scoring of speech using different representation approaches.

Fourth, this research made a thorough analysis of different representation approaches and presented a detailed picture of how these approaches worked. In addition to comparing approaches based on performance measurements, the author also conducted vector and case analyses to obtain an in-depth understanding of vector dimensions and transcript content.

Fifth, these representation approaches along with the reasoning method can be applied to other domains. In fact, this workflow, first representing and then machine learning, is applicable to any information related tasks that can be modeled as a text classification task. This workflow can potentially be a workbench for examining the effect of different representations on different tasks. One domain that can immediately apply the methodology is automatic essay scoring. As mentioned in section 5.3, longer text with less noise in essays can generate better representation output by using ontology-based representation and more likely achieve a better performance than in speech scoring.

Lastly, the study also contributes to a potentially standardized representation of text documents. Text documents are usually considered as plain text files and strings, and the embedded semantics can only be consumed by human reading them. This study considers text as an object that can have structures, properties, and sub-

elements. This study showcased that some types of vector representations of text could be used as a standard way of text representation along with other possible ones. From an ontological perspective, if a text document is treated as an object, this object will be a vector representation and contain words, synsets or Wikipedia concepts, which are sub-elements of the text object and can be semantically linked to each other to form a semantic network. These relations and instances can be conveniently expressed in standard Semantic Web languages such as RDF (Resource Description Dramework) and OWL (Web Ontology Language) to better share and reuse the text object. In addition, the various vectors in this study are potentially useful for representing text content as objects using semantic web languages.

5.5 Limitations

This study has seven limitations. First, although significance test can provide more convincing comparison between approaches, such test is not conducted widely in the result analysis due to the small number of prompts (n=4). The current comparison analysis between approaches is to compare their average performance over the 4 prompts. Though with a small sample size, the author have made an attempt to run a paired sample t-test between BOW(tfidf) and Combined (BOW=0.7, Wn1st=0.3), the result of which turn to be not conclusive to tell whether the Combined (BOW=0.7, Wn1st=0.3) can enhance the performance significantly over BOW(tfidf). The data size affects the significance test results and thus it is difficult to draw conclusions based on the current sample size. If there were more prompts, significance tests such as t-tests and ANOVA can be run to more rigorously examine whether performance differences between approaches are statistically significant.

Second, related to the first point, the conclusions would be more reliable with a larger data set containing more prompts and speech transcripts. Domingo (2012) points out that a larger data set with a dumb algorithm beats a smaller data set with a clever algorithm, indicating the importance of large data size. Given the fact that the data set of this study has 1237 speech transcripts, it explains why the scoring module performance is less than ideal. It even caused Naïve Bayes machine learning model function poorly due to data scarcity. A larger data set would have not only eliminated the data scarcity problem, but also helped better utilize the power of ontology in text representation. As discussed in section 4.3.2, one reason of low performance when only using WordNet concepts in vectors is that there are not sufficient synonymous words in the corpus to allow the ontology-based representation to merge them in the same dimension. A larger data set would have increased the possibility of having more synonymous words and extended the benefit of ontology-based representation to a greater extent.

Third, the speech construct contains various factors besides content. The most accurate scoring model should consist of factors of different aspects, however this study only included content vectors in the scoring model. It is expected that by integrating acoustic features the predicting model will achieve a better performance, as content only contributes to part of the scoring. Because the model in this study only contains content vectors, the performance is reasonable when compared to the performance range of un-transformed features in SpeechRater. The highest max.coso correlations in this study, 0.3704 from Combined (BOW=0.7, Wn1st=0.3), compares quite favorably to the correlations in Zechner et al.'s (2009) study where 4 of 5 feature correlations are in the range of [0.10, 0.45].

Fourth, this study used human-assigned holistic scores to evaluate the content of spoken responses. Since these holistic scores are based on several sub-dimensions, such as fluency, vocabulary, grammar, as well as content, changes in content representation may have less of an effect than if there were human scores available that evaluate the content aspect of the construct exclusively.

Fifth, from text and document side, certain amount of information is missing in the current representation. First, word or concept order in the text was not preserved. Order can be important in scoring because it entails language competence of speakers, such as organizational competence in Bachman's (1990) model, a type of competence about organizing words and sentences. Second, linguistic annotations were not present in the representation. Although part-of-speech annotation was used in the Wnpos approach for synset matching, annotations were not used extensively in the representations in this study. Annotation information includes phrases, parsing trees, part-of-speech tags, semantic role labeling, and recognized entities, which obviously contain rich information for the text. Recently, Clarke et al. (2012) published a toolkit that aggregates various types of NLP annotations, which is an example of representing text by different annotations. Going beyond representing text using words and concepts in the vector style can further enhance representation and possibly improve classification performance, if done in an appropriate way.

Sixth, the scoring models are supposed to classify the levels of content quality instead of classifying topics of content. Text vectors are usually good for representing topics because different topics use different words, but this is not always true for content quality classification. A highly-scored speech would contain words relevant to the

prompt's topic while a lowly-scored speech may also contain such words to a large extent. This characteristic of content quality makes it difficult to classify speech transcripts by content features only. This also explains why the performance of all approaches seems relatively low compared to some acoustics features, e.g. transformed Amscore which has a 0.510 correlation with human raters in Zechner et al. (2009).

Seventh, representation approaches was evaluated indirectly through the performance of content scoring rather than the characteristics of themselves, for instance, evaluating based on the vectors. Thus far the representations in other studies (e.g. Lewis, 1990; Scott & Matwin, 1999) are also evaluated indirectly, since the ultimate goal of representation lies in facilitating text related tasks. However, some systematic means can be used to directly evaluate representation, e.g. the number of vector dimensions and the number of unknown concepts in test set.

5.6 Future Work

There are several directions for future work. First, within the same data set, ontology-based approaches can be further experimented by using other parameter setup. For example, we can perform feature selection on concepts, add hypernyms of WordNet synsets to the vector, or rerank Wikipedia concepts from the ESA representation.

Second, e-rater, a similarity-based classification method, is the primary machine learning model in this study. The main idea is that a test transcript is classified to the score level that has the largest similarity to it. Because the score level is an ordinal

class, future research can revise the classification algorithms designed for ordinal values to make use of the ordinal information in the data set.

Lastly, scientific computation has entered into the Big Data era, featuring in its variety, velocity, and volume and data (TechAmerica Foundation's Big Data Commission, 2012). One trend of Big Data is that more and more data sets are becoming openly accessible and forming a linked data space. Since the study is limited by its small data set, this may provide an opportunity for acquiring related data sets for the automatic scoring task.

REFERENCES

Arguello, J., Elsas, J. L., Callan, J., & Carbonell, J. G. (2008). *Document representation and query expansion models for blog recommendation.* In *Proceedings of the 2nd International Conference on Weblogs and Social Media (ICWSM 2008).*

Attali, Y., & Burstein, J. (2006). Automated essay scoring with e-rater® V. 2. *The Journal of Technology, Learning and Assessment, 4*(3).

Attardi, G., & Fuschetto, A. (2013). Wikipedia Extractor (Version 2.4) [Software]. Available from http://medialab.di.unipi.it/wiki/Wikipedia_Extractor

Bachman, L. F. (1990). *Fundamental considerations in language testing*: Oxford University Press, USA.

Bachman, L. F. (2004). *Statistical analyses for language assessment*: Cambridge University Press.

Bachman, L. F., & Palmer, A. S. (1996). *Language testing in practice: Designing and developing useful language tests*: Oxford University Press.

Banerjee, M., Capozzoli, M., McSweeney, L., & Sinha, D. (1999). Beyond kappa: A review of interrater agreement measures. *Canadian Journal of Statistics*, *27*(1), 3-23.

Bagarić, V., & Djigunović, J. M. (2007). Defining communicative competence. *METODIKA: časopis za teoriju i praksu metodika u predškolskom odgoju, školskoj i visokoškolskoj izobrazbi*, *8*(14), 94-103.

Bernstein, J., Van Moere, A., & Cheng, J. (2010). Validating automated speaking tests. *Language Testing, 27*(3), 355-377.

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with Python*. O'Reilly Media.

Bisson, G., Nedelee, C., Canamero, D. (2000). *Designing clustering methods for ontology building: The Mo'K workbench*. *The 1st Workshop on Ontology Learning in conjunction with the 14th European Conference on Artificial Intelligence* (pp.13-19).

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., & Hellmann, S. (2009). DBpedia-A crystallization point for the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web, 7*(3), 154-165.

Blei, D. (2011). Introduction to probabilistic topic models. *Communications of the ACM*, *55*(4), 77-84.

Bloehdorn, S., Blohm, S., Cimiano, P., Giesbrecht, E., Hotho, A., Lösch, U., ... & Völker, J. (2011). Combining Data-Driven and Semantic Approaches for Text Mining. In *Foundations for the Web of Information and Services* (pp. 115-142). Springer Berlin Heidelberg.

Bloehdorn, S., & Hotho, A. (2004). Boosting for text classification with semantic features. *The workshop on mining for and from the semantic web at the 10th ACM SIGKDD conference on knowledge discovery and data mining (KDD 2004)*.

Bradford, R. B. (2008, October). An empirical study of required dimensionality for large-scale latent semantic indexing applications. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management* (pp. 153-162), New York, NY, USA. ACM.

Buckland, M. K. (1991). Information as thing. *Journal of the American Society for information science, 42*(5), 351-360.

Burstein, J. (2003). The E-rater® scoring engine: Automated essay scoring with natural language processing. In M. D. Shermis, Burstein, J.C. (Ed.), *Automated essay scoring: A cross-disciplinary perspective* (pp. 113-121). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Canale, M. (1983). From communicative competence to communicative language pedagogy. In Richards, J. C., & Schmidt, R. W. (Eds.), *Language and Communication*, 2-27. London: Longman.

Canale, M., & Swain, M. (1980). Theoretical bases of com-municative approaches to second language teaching and testing. *Applied linguistics, 1*(1), 1-47.

Chen, L., Tokuda, N., & Nagai, A. (2003). A new differential LSI space-based probabilistic document classifier. *Information Processing Letters, 88*(5), 203-212.

Chen, L., Zeng, J., & Tokuda, N. (2006). A “stereo” document representation for textual information retrieval. *Journal of the American Society for Information Science and Technology, 57*(6), 768-774.

Chen, M., & Zechner, K. (2011). Computing and evaluating syntactic complexity features for automated scoring of spontaneous non-native speech. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics and the Human Language Technologies Conference* (pp.722-731), Portland, Oregon.

Chomsky, N. (1965). *Aspects of the Theory of Syntax*. Cambridge, MA: The MIT press.

Chu, H. (2003). *Information representation and retrieval in the digital age*. Medford, NJ: Information Today Inc.

Cilibrasi, R. L., & Vitanyi, P. M. B. (2007). The google similarity distance. *Knowledge and Data Engineering, IEEE Transactions on, 19*(3), 370-383.

Clarke, D. (2012). A context-theoretic framework for compositionality in distributional semantics. *Computational Linguistics*, *38*(1), 41-71.

Clarke, J., Srikumar, V., Sammons, M., & Roth, D. (2012). An nlp curator (or: How i learned to stop worrying and love nlp pipelines). In *Proceedings of the International Conference on Language Resources and Evaluation 2012* (pp.3276-3283), Istanbul, Turkey.

Croft, W. B., Metzler, D., & Strohman, T. (2010). *Search engines: Information retrieval in practice*. Boston, MA: Addison-Wesley.

Cucchiarini, C., Strik, H., & Boves, L. (2000). Quantitative assessment of second language learners' fluency by means of automatic speech recognition technology. *The Journal of the Acoustical Society of America, 107*(2), 989-999.

Cucchiarini, C., Strik, H., & Boves, L. (2002). Quantitative assessment of second language learners' fluency: Comparisons between read and spontaneous speech. *the Journal of the Acoustical Society of America*, *111*, 2862.

De Jong, N. H., Steinel, M. P., Florijn, A. F., Schoonen, R., & Hulstijn, J. H. (2011). Facets of speaking proficiency. *Studies in Second Language Acquisition, 34,* 4-34.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for information science, 41*(6), 391-407.

Dikli, S. (2006). An overview of automated scoring of essays. *The Journal of Technology, Learning and Assessment, 5*(1), 3-35.

Dodigovic, M. (2009). Speech Processing Technology in Second Language Testing. In *Proceedings of the Conference on Language & Technology 2009* (pp. 113-120).

Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM*, *55*(10), 78-87.

Fellbaum, C. (Ed.). (1998). *WordNet: An electronic lexical database*. Cambridge, MA: The MIT press.

Finlayson, M. A. (2012). JWI (the MIT Java WordNet Inteface) (Version 2.2.3) [Software]. Available from http://projects.csail.mit.edu/jwi/

Foltz, P. W., Laham, D., & Landauer, T. K. (1999). *Automated essay scoring: Applications to educational technology.* In *World Conference on Educational Multimedia, Hypermedia and Telecommunications* (Vol. 1999, No. 1, pp. 939-944).

Gabrilovich, E., & Markovitch, S. (2007). Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence* (Vol. 7, pp. 1606-1611).

Gabrilovich, E. (2007, December). *Feature generation for textual information retrieval using world knowledge* (Doctoral dissertation). Retrieved Aug 20, 2013 from http://www.cs.technion.ac.il/~gabr/papers/phd-thesis.pdf

Gabrilovich, E., & Markovitch, S. (2009). Wikipedia-based semantic interpretation for natural language processing. *Journal of Artificial Intelligence Research, 34*(1), 443-498.

Garcia, E. (2006). Latent Semantic Indexing (LSI) A Fast Track Tutorial. Retrieved Aug 20, 2013, from http://www.aplusmarketingservices.com/wp-content/uploads/2012/05/latent-semantic-indexing-fast-track-tutorial.pdf

Gómez-Pérez, A., & Corcho, O. (2002). Ontology Languages for the Semantic Web. *IEEE Intelligent Systems,* 17 *(*1), 54-60.

Gottron, T., Anderka, M., & Stein, B. (2011, October). Insights into explicit semantic analysis. In *Proceedings of the 20th ACM international conference on Information and knowledge management* (pp. 1961-1964). ACM.

Gruber, T. What is an ontology? Retrieved Aug 20, 2013, from http://www-ksl.stanford.edu/kst/what-is-an-ontology.html

Gruber, T. (1993). A translation approach to portable ontologies. *Knowledge Acquisition, 5*(2), 199-220.

Gruber, T. (2009). What is an Ontology? In L. Liu & M. T. Özsu (Eds.), *Encyclopedia of Database Systems*: Springer-Verlag.

Gruninger, M., & Fox, M. (1995). Methodology for the design and evaluation of ontologies. In *Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing at International Joint Conference on Articial Intelligence 1995*.

Guarino, N. (1998). Formal ontology in information systems. In N. Guarino (Ed.), *Proceedings of the Formal Ontology in Information Systems 1998 (FOIS '98)*. Trento, Italy: Ios Pr Inc.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD Explorations Newsletter, 11*(1), 10-18.

He, X., Cai, D., Liu, H., & Ma, W. Y. (2004). Locality preserving indexing for document representation. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 96-103). ACM.

Hope, D. (2008). Java WordNet::Similarity (beta 11.01) [Software]. Available from http://www.sussex.ac.uk/Users/drh21/

Hotho, A., Staab, S., & Stumme, G. (2003a). Ontologies improve text document clustering. In *Proceedings of the Third IEEE International Conference on Data Mining 2003* (pp. 541-544).

Hotho, A., Staab, S., & Stumme, G. (2003b). *Text clustering based on background knowledge* (Technical report, no.425.): Institute of Applied Informatics and Formal Description Methods AIFB, University of Karlsruche.

How the Versant Testing System Works. Retrieved Aug 20, 2013, from http://www.ordinate.com/technology/scoring.jsp

IELTS. (2013). Retrieved Aug 20, 2013 from http://en.wikipedia.org/wiki/IELTS

Iwashita, N., Brown, A., McNamara, T., & O'Hagan, S. (2008). Assessed levels of second language speaking proficiency: How distinct?. *Applied Linguistics, 29*(1), 24-49.

Joachims, T. (1997). A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. In *Proceedings of the Fourteenth International Conference on Machine Learning* (pp. 143–151), San Francisoco, CA. Morgan Kaufman.

Johansson, I. (2004). *Ontological investigations: An inquiry into the categories of nature, man, and society* (2nd ed.). Frankfurt: Ontos Verlag.

Kaski, S. (1997). Computationally efficient approximation of a probabilistic model for document representation in the WEBSOM full-text analysis method. *Neural processing letters, 5*(2), 69-81.

Kaski, S., Honkela, T., Lagus, K., & Kohonen, T. (1998). WEBSOM-Self-organizing maps of document collections1. *Neurocomputing, 21*(1-3), 101-117.

Kontostathis, A., & Pottenger, W. M. (2006). A framework for understanding Latent Semantic Indexing (LSI) performance. *Information Processing and Management, 42*, 56-73.

Landauer, T. K., Laham, D., & Foltz, P. W. (2003). Automated scoring and annotation of essays with the Intelligent Essay Assessor. In M. D. Shermis, Burstein, J.C. (Ed.), *Automated essay scoring: A cross-disciplinary perspective* (pp. 87–112). Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Landauer, T. K., Laham, D., Rehder, B., & Schreiner, M. (1997). How well can passage meaning be derived without using word order? A comparison of Latent Semantic Analysis and humans. In *Proceedings of the 19th Annual Meeting of the Cognitive Science Society* (pp. 412-417).

Larkey, L. S., & Croft, W. B. (2003). A Text Categorization Approach to Automated Essay Grading. In M. D. Shermis & J. C. Burstein (Eds.), *Automated Essay Scoring: A Cross-discipline Perspective*: Mahwah, NJ, Lawrence Erlbaum.

Levelt, W. J. M. (1989). *Speaking: from intention to articulation*. Cambridge, MA: MIT Press.

Levelt, W. J. M., Roelofs, A., & Meyer, A. S. (1999). A theory of lexical access in speech production. *Behavioral and Brain Sciences*, 22(1), 1 - 37.

Lewis, D. (1990, June). Representation quality in text classification: An introduction and experiment. In *Proceedings of Workshop on Speech and Natural Language. Morgan Kaufmann, Hidden Valley, PA* (pp. 288-295).

Lewis, D. D. (1992). *Representation and learning in information retrieval* (Doctoral dissertation). Retrieved from ProQuest Dissertations and Theses database (Order No. 9219460, University of Massachusetts Amherst).

Lin, D. (1998). An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning* (Vol. 98, pp. 296-304).

Lin, X., Soergel, D., & Marchionini, G. (1991). A self-organizing map for information retrieval. In *Proceedings of the 14th Annual International ACM/SIGIR Conference on R&D in Information Retrieval* (pp. 262-269). ACM.

Linckels, S., & Meinel, C. (2011). *E-librarian Service: User-friendly Semantic Search in Digital Libraries*. Springerverlag Berlin Heidelberg.

Mani, I., Samuel, K., Concepcion, K., & Vogel, D. (2004). Automatically inducing ontology from corpora. In *Proceedings of CompuTerm 2004: 3rd International Workshop on Computational Terminology, COLING'2004,* Geneva.

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

Milne, D., & Witten, I. H. (2008). An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In *AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy, AAAI Press,* Chicago, USA (pp. 25-30).

Muller, H. M., Kenny, E. E., & Sternberg, P. W. (2004). Textpresso: an ontology-based information retrieval and extraction system for biological literature. *PLoS Biology, 2*(11), e309.

Neches, R., Fikes, R.E., Finin, T., Gruber, T.R., Senator, T., Swartout, W.R. (1991). Enabling technology for knowledge sharing. *AI Magazine, 12*(3), 36-56.

Noy, N. F., & McGuinness, D. L. (2001). Ontology Development 101: A Guide to Creating Your First Ontology. *Stanford Knowledge System Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880*.

OpenNLP (Version 1.5.2) [Software] (2011). Available from http://opennlp.apache.org

Ordinate. (2005). *SET-10: Test description – Validation summary*. Menlo Park, Harcourt.

Özgür, A., Özgür, L., & Güngör, T. (2005). Text categorization with class-based and corpus-based keyword selection. In *Computer and Information Sciences-ISCIS 2005* (pp. 606-615). Springer Berlin Heidelberg.

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, *2*(1-2), 1-135.

Pedersen, T., Patwardhan, S., Banerjee, S., & Michelizz, J. (2008). WordNet::Similarity [Software]. Available from http://wn-similarity.sourceforge.net/

Pedersen, T., Patwardhan, S., & Michelizzi, J. (2004). WordNet:: Similarity: measuring the relatedness of concepts. In *Proceedings of the Fifth Annual Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL-04)* (pp. 38-41), Boston, MA.

Ratinov, L., Roth, D., Downey, D., & Anderson, M. (2011, June). Local and global algorithms for disambiguation to Wikipedia. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL 2011)* (pp. 1375-1384).

Řehůřek, R., & Sojka, P. (2010). Software framework for topic modelling with large corpora. In *Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks* (pp. 46-50).

Resnik, P. (1999). Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence, 11*(1999), 95-130.

Rocchio, J. J. (1971). Relevance feedback in information retrieval. In Salton (Eds.), *The SMART Retrieval System: Experiments in Automatic Document Processing* (pp. 313-323), Prentice-Hall.

Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM, 18*(11), 613-620.

Santorini, B. (1990). *Part-of-speech tagging guidelines for the Penn Treebank Project* (Technical report MS-CIS-90-47). Department of Computer and Information Science, University of Pennsylvania.

Schmitz, P. (2006). Inducing Ontology from Flickr Tags. In *Proceedings of the Collaborative Web Tagging Workshop at the World Wide Web Conference 2006*.

Scott, S., & Matwin, S. (1999). Feature engineering for text classification. In *Proceedings of the 16th International Conference on Machine Learning ICML-99* (pp. 379-388).

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR), 34*(1), 1-47.

Sowa, J. F. (2000). *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Pacific Grove, CA: Brooks/Cole.

Strube, M., & Ponzetto, S. P. (2006). WikiRelate! Computing semantic relatedness using Wikipedia. In *Proceedings of the American Association for Artificial Intelligence 2006* (pp. 1419-1424), Boston, MA.

Swartout, B., Patil, R., Knight, K., & Russ, T. (1996). Toward Distributed Use of Large-Scale Ontologies. In *Proceedings of the Tenth Workshop on Knowledge Acquisition for Knowledge-Based Systems* (pp. 32.1-32.19), Alberta, Canada.

TechAmerica Foundation's Big Data Commission. (2012). *Demystifying big data: A practical guide to transforming the business of government.* Retrieved Aug 20, 2013, from http://www.techamerica.org/Docs/fileManager.cfm?f=techamerica-bigdatareport-final.pdf

Tengi, R. I. (1998). Design and implementation of the WordNet lexical database and searching software. In C. Fellbaum (Ed.), *WordNet: An electronic lexical database* (pp. 105-127). Cambridge, MA: The MIT Press.

Valenti, S., Neri, F., & Cucchiarelli, A. (2003). An overview of current research on automated essay grading. *Journal of Information Technology Education, 2*, 319-330.

Versant. (2013). In *Wikipedia*. Retrieved Apr 20, 2013, from http://en.wikipedia.org/wiki/Versant

Wang, B. B., McKay, R. I., Abbass, H. A., & Barlow, M. (2003). A comparative study for domain ontology guided feature extraction. In *Proceedings of the 25th Australian*

Computer Science Conference – Volume 16 (pp. 69-78), Adelaide, Australia. Australian Computer Society, Inc..

Wikipeida. (2013). In *Wikipedia*. Retrieved Aug 20, 2013 from http://en.wikipedia.org/wiki/Wikipedia

WordNet. (n.d.). In *Wikipedia*. Retrieved Aug 20, 2013, from http://en.wikipedia.org/wiki/Wordnet

Wordnet statistics (n.d.). Retrieved Aug 20, 2013 from http://wordnet.princeton.edu/wordnet/man/wnstats.7WN.html

WordNet database files. Retrieved Aug 20, 2013, from http://wordnet.princeton.edu/wordnet/man/wndb.5WN.html

Xi, X., Higgins, D., Zechner, K., & Williamson, D. M. (2008). Automated scoring of spontaneous speech using SpeechRaterSM v1. 0. *Educational Testing Services Research Report*.

Xie, S., Evanini, K., & Zechner, K. (2012, June). Exploring content features for automated speech scoring. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 103-111). Association for Computational Linguistics.

Yang, Y., & Pedersen, J. O. (1997, July). A comparative study on feature selection in text categorization. In *Proceedings of 14th International Conference on Machine Learning (pp.412-420).*

Zechner, K., Higgins, D., Xi, X., & Williamson, D. M. (2009). Automatic scoring of non-native spontaneous speech in tests of spoken English. *Speech Communication, 51*(10), 883-895.

Zhang, X. (2009). *Exploiting External/Domain Knowledge to Enhance Traditional Text Mining Using Graph-based methods*. (Doctoral dissertation). Retrieved from ProQuest Dissertation and Theses. (AAT 3361939)

Appendix 1

Mapping between WordNet senses and Penn Treebank POS tags

Penn TreeBank	WordNet sense
NN	n
NNS	n
NNP	n
NNPS	n
VB	v
VBD	v
VBG	v
VBN	v
VBP	v
VBZ	v
JJ	a,s
JJR	a,s
JJS	a,s
RB	r
RBR	r
RBS	r
WRB	r

Meaning of the Penn TreeBank tags (Santorini, 1990):

NN=Noun, singular or mass

NNS=Noun, plural

NNP=Proper noun, singular

NNPS=proper noun, plural

VB=verb, base form

VBD=verb, past tense

VBG=verb, gerund or present participle

VBN=verb, past participle

VBP=Verb, non-3rd person singular present

VBZ=verb, 3rd person singular present

JJ=adjective

JJR=adjective, comparative

JJS=adjective, superlative

RB=adverb

RBR=adverb, comparative

RBS=adverb, superlative

WRB=Wh-adverb

Meaning of the WordNet types (“WordNet database files,” 2011):

n=noun

v=verb

a= adjective

s=satellite adjective (simlar synsets to the head synset of a synset cluster)

r=adverb

Appendix 2

Details of the 4 TPO 2006 prompts, provided by Educational Testing Service.

Prompt 098

N City University is planning to increase tuition and fees. Read the announcement about the increase from the president of City University. You will have 45 seconds to read the announcement. Begin reading now.

Announcement from the president

The university has decided to increase tuition and fees for all students by approximately 8 percent next semester. For the past 5 years, the tuition and fees have remained the same, but it is necessary to increase them now for several reasons. The university has many more students than we had five years ago, and we must hire additional professors to teach these students. We have also made a new commitment to research and technology, and will be renovating and upgrading our laboratory facilities to better meet our students' needs.

N Now listen to two students as they discuss the announcement. 2 seconds

MB-AP Oh great, now we have to come up with more money for next semester.

WB-KS Yeah, I know, but I can see why. When I first started here, classes were so much smaller than they are now. With this many students, it's hard to get the personal attention you need...

MB Yeah, I guess you're right. You know, in some classes I can't even get a seat. And I couldn't take the math course I wanted to because it was already full when I signed up.

WB And the other thing is, well, I am kind of worried about not being able to get a job after I graduate.

MB Why? I mean you're doing really well in your classes, aren't you?

WB I'm doing ok, but the facilities here are so limited. There are some great new experiments in microbiology that we can't even do here... there isn't enough equipment in the laboratories, and the equipment they have is out of date. How am I going to compete for jobs with people who have practical research experience? I think the extra tuition will be a good investment.2 seconds

N The woman expresses her opinion of the announcement made by the university president. State her opinion and explain the reasons she gives for holding that opinion.

Prompt 099

N Now read the passage about animal domestication. You have 45 seconds to read the passage. Begin reading now.

Animal Domestication

For thousands of years, humans have been able to domesticate, or tame, many large mammals that in the wild live together in herds. Once tamed, these mammals are used for agricultural work and transportation. Yet some herd mammals are not easily domesticated.

A good indicator of an animal's suitability for domestication is how protective the animal is of its territory. Non-territorial animals are more easily domesticated than territorial animals because they can live close together with animals from other herds. A second indicator is that animals with a hierarchical social structure, in which herd

members follow a leader, are easy to domesticate, since a human can function as the "leader".

N Now listen to part of a lecture on this topic in an ecology class.2 seconds

MA-IA So we've been discussing the suitability of animals for domestication... particularly animals that live together in herds. Now, if we take horses, for example... in the wild, horses live in herds that consist of one male and several females and their young. When a herd moves, the dominant male leads, with the dominant female and her young immediately behind him. The dominant female and her young are then followed immediately by the second most important female and her young, and so on. This is why domesticated horses can be harnessed one after the other in a row. They're "programmed" to follow the lead of another horse. On top of that, you often find different herds of horses in the wild occupying overlapping areas--they don't fight off other herds that enter the same territory.

But it's exactly the opposite with an animal like the uh, the antelope... which... well, antelopes are herd animals too. But unlike horses, a male antelope will fight fiercely to prevent another male from entering its territory during the breeding season, ok--very different from the behavior of horses. Try keeping a couple of male antelopes together in a small space and see what happens. Also, antelopes don't have a social hierarchy--they don't instinctively follow any leader. That makes it harder for humans to control their behavior.2 seconds

N

The professor describes the behavior of horses and antelope in herds. Explain how their behavior is related to their suitability for domestication.

Prompt 100

Now listen to a conversation between two students.

MB--AP Hey Lisa, how's it going?

WB--KM Hi Mark. Uh, I,Äôm OK, I guess, but my schoolwork is really stressing me out.

MB [sympathetically] Yeah? What's wrong?

WB Well, I,Äôve got a paper to write, and two exams to study for. And a bunch of math problems to finish. It's just so much that I can,Äôt concentrate on any of it. I start concentrating on studying for one of my exams, and then I'm like, how long's it gonna take to finish that problem set?

MB Wow sounds like you've got a lot more work than you can handle right now. [Not wanting to sound too pushy] Look have you talked to some of your professors...I mean, you know , try to explain the problem. Look, you could probably get an extension on your paper, or on the math assignment...

WB You think? It would give me a little more time to prepare for my exams right now.

MB Well, I mean another thing that you might do ... I mean have you tried making yourself a schedule? I mean that's what I do when I,Äôm feeling overwhelmed.

WB What does that do for you?

MB Well, I mean it helps you to focus your energies. You know, you make yourself a chart that shows the next few days and the time till your stuff is due and...

WB Uh-huh [meaning "I'm listening"]

MB I mean think about what you need to do, and when you have to do it by. You know then start filling in your schedule--like, all right 9:00 [nine] to 11:30 [eleven-thirty] A.M., study for exam. 12:00 [twelve] to 3:00 [three], work on problem set. But I mean don't make the time periods too long. Like, don't put in eight hours of studying--you know, you'll get tired, or start worrying about your other work again. But if you keep to your schedule, you know you,Äôll just have to worry about one thing at a time.

WB Yeah, that might work. [somewhat noncommitally]

N The students discuss two possible solutions to the woman's problem. Describe the problem. Then state which of the two solutions you prefer and explain why.

Prompt 101

Now listen to part of a talk in a United States history class.

WA--MM Because the United States is such a large country, it took time for a common national culture to emerge. A hundred years ago there was very little communication among the different regions of the United States. One result of this lack of communication was that people around the United States had very little in common with one another. People in different parts of the country spoke differently, dressed differently, and behaved differently. But connections among Americans began to increase thanks to two technological innovations: the automobile and the radio.

Now automobiles began to be mass produced in the 1920's, which meant they became less expensive and more widely available. Americans in small towns and rural communities now had the ability to travel easily to nearby cities. They could even take vacations to other parts of the country. This increased mobility that automobiles provided changed people's attitudes and created links that hadn't existed before. For

example, people in small towns began to adopt behaviors, clothes, and speech that were popular in big cities or in other parts of the country. As more Americans were purchasing cars, radio ownership was also increasing dramatically. Americans in different regions of the country began to listen to the same popular radio programs and the same musical artists. People repeated things they heard on the radio--some phrases and speech patterns they heard in songs and on radio programs began to be used by people all over the United States. People also listened to news reports on the radio. So they heard the same news throughout the country, whereas in newspapers much of the news tended to be local. So radio brought Americans together by offering them shared experiences and information about events all around the country.

N Using points and examples from the talk, explain how the automobile and the radio contributed to a common culture in the United States.

Appendix 3

It shows a complete Table of performance of the representation approaches.

Approach	Experiment	Avg. max.cos corr.	Avg. cos.w4 corr.	Avg. F measure	Avg. accuracy	Avg. kappa
BOW	BOW(tfidf)	0.3494	0.3556	0.4627	0.4786	0.3441
BOW	BOW(tf)	0.2259	0.0806	0.3804	0.3789	0.2178
LSA	LSA (k=10)	0.1751	0.1618	0.3533	0.4116	0.1496
LSA	LSA (k=20)	0.2242	0.0823	0.3848	0.4204	0.2039
LSA	LSA (k=30)	0.2003	0.1016	0.3727	0.3852	0.1927
LSA	LSA (k=40)	0.2506	0.151	0.3931	0.4442	0.2393
LSA	LSA (k=50)	0.228	0.1053	0.412	0.445	0.2225
LSA	LSA (k=100)	0.2394	0.1395	0.368	0.3898	0.2272
LSA	LSA (k=200)	0.1998	0.1791	0.3664	0.392	0.1887
ONTO-WordNet	Wn1st	0.2686	0.3478	0.4422	0.4595	0.2656
ONTO-WordNet	Wnpos	0.2494	0.3281	0.4398	0.4662	0.2469
ONTO-WordNet	Combined (Wn1st, BOW)	0.343	0.3653	0.4631	0.4815	0.3382
ONTO-WordNet	Combined (Wnpos, BOW)	0.3323	0.3588	0.4577	0.4796	0.3272
ONTO-WordNet	Combined (Wn1st repl BOW)	0.2957	0.2403	0.4371	0.4542	0.2923
ONTO-WordNet	Combined (Wnpos repl BOW)	0.3053	0.331	0.4495	0.471	0.3018
ONTO-Wikipedia	DirectWiki	0.1444	0.187	0.3489	0.3749	0.1366
ONTO-Wikipedia	ESA (n=10)	0.0469	0.0483	0.3023	0.3302	0.0413
ONTO-Wikipedia	ESA (n=20)	0.0565	0.05	0.3199	0.3429	0.0515
ONTO-Wikipedia	ESA (n=50)	0.0789	0.0548	0.345	0.3583	0.075
ONTO-Wikipedia	ESA (n=100)	0.0467	0.051	0.3391	0.3606	0.0431
ONTO-Wikipedia	ESA (n=1000)	0.0183	0.0503	0.3139	0.3576	0.0156
ONTOReason-WordNet	WNreasoning (Wn1st, Path)	0.2511	0.372	0.4342	0.4374	0.2486
ONTOReason-WordNet	WNreasoning (Wn1st, Lin)	0.2266	0.3769	0.4241	0.4265	0.2236
ONTOReason-WordNet	WNreasoning (Wn1st, Dft)	0.2422	0.3864	0.4246	0.4249	0.2381
ONTOReason-WordNet	WNreasoning (Wnpos, Path)	0.2153	0.3543	0.4213	0.4253	0.2123
ONTOReason-WordNet	WNreasoning (Wnpos, Lin)	0.2119	0.3667	0.4135	0.4155	0.2076
ONTOReason-WordNet	WNreasoning (Wnpos, Dft)	0.2176	0.3709	0.412	0.413	0.2138
ONTOReason-	WikiReasoning(Content)	0.1217	0.1929	0.3336	0.3469	0.1124

Wikipedia						
ONTOReason-Wikipedia	WikiReasoning(Dft)	0.1343	0.1958	0.3246	0.3413	0.1245
Combined	Combined (BOW, Wn1st, Wiki)	0.2268	0.3244	0.3978	0.4203	0.2204
Combined	Combined (BOW, Wnpos, Wiki)	0.2154	0.3201	0.3875	0.4129	0.2086
Combined	Combined (BOW, esa10)	0.3582	0.2895	0.4682	0.4789	0.3548
Combined	Combined (BOW, esa20)	0.3245	0.2627	0.4516	0.4608	0.3209
Combined	Combined (BOW, esa50)	0.3431	0.1935	0.4587	0.4675	0.3342
Combined	Combined (BOW, esa100)	0.3282	0.1304	0.4324	0.4478	0.3075
Combined	Combined (BOW, esa1000)	0.2278	-0.0699	0.3469	0.3778	0.1791
Combined	Combined (BOW=0.7, Wn1st=0.3)	0.3704	0.3651	0.4710	0.4869	0.3648
Combined	Combined (BOW=0.75, Wn1st=0.25)	0.3699	0.3643	0.4709	0.4860	0.3642
Combined	Combined (BOW=0.6, Wn1st=0.4)	0.3634	0.3658	0.4690	0.4863	0.3583
Combined	Combined (BOW=0.7, Wn1st=0.2, wiki=0.1)	0.3511	0.3797	0.4667	0.4851	0.3464
Combined	Combined (BOW=0.7, Wn1st=0.25, wiki=0.05)	0.3605	0.3773	0.4699	0.4877	0.3552
Combined	Combined (BOW=0.6, Wn1st=0.2, wiki=0.2)	0.3100	0.3638	0.4401	0.4607	0.3044
Combined	Combined (BOW=0.7, Wn1st=0.1, wiki=0.2)	0.3061	0.3613	0.4394	0.4591	0.3007

VITA

Name of Author: Miao Chen

EDUCATION

Bachelor of Science, Peking University, China 2005
Bachelor in Economics, Peking University, China 2005

PROFESSIONAL EXPERIENCE

Visiting Research Associate (2011-2013)
Data To Insight Center, Indiana University Bloomington, IN

Research Intern (2010, 2011)
Educational Testing Service, Princeton, NJ

Software Engineering Intern (2009)
Progressive Expert Consulting, Syracuse, NY

AWARDS

Syracuse University Fellowship (2005-2009)

KEY PUBLICATIONS

Chen, M. & Zechner, K. (2012). Using an Ontology for Improved Automated Content Scoring of Spontaneous Non-native Speech. The 7th Workshop on Innovative Use of NLP for Building Educational Applications at the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2012).

Chen, M., & Zechner, K. (2011). Computing and Evaluating Syntactic Complexity Features for Automated Scoring of Spontaneous Non-Native Speech. The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT 2011). Portland, OR.

Chen, M., Liu, X., & Qin, J. (2008). Semantic relation extraction from socially-generated Tags: A methodology for metadata generation. The 8th International Conference on Dublin Core and Metadata Applications (DC 2008). Berlin, Germany.