5-1-2018

IMPROVING AUTOMATIC SUMMARIZATION FOR LOW- AND MODERATE-RESOURCE, MORPHOLOGICALLY COMPLEX LANGUAGES

Kalen Goss Manshack

IMPROVING AUTOMATIC SUMMARIZATION FOR LOW- AND MODERATE-RESOURCE, MORPHOLOGICALLY COMPLEX LANGUAGES

by

KALEN GOSS MANSHACK

Presented to the Faculty of the Honors College of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

HONORS BACHELOR OF ARTS IN LINGUISTICS

THE UNIVERSITY OF TEXAS AT ARLINGTON

May 2018

ACKNOWLEDGMENTS

First, I would like to thank my thesis mentor and the professor who introduced me to an entire career field I never knew and to people who have helped me greatly along the way, Dr. Pete Smith. He has taken a lot of time out of his already extremely busy schedule to help me in my academic and professional career, and I am extremely thankful.

I would also like to thank Dr. Mike Dillinger. He has been my patient and kind sounding board over the last year and has spent so much of his time helping me when he had absolutely no obligation to. He answered my endless questions about the details of my project and thesis, and always steered my thinking back to the right path. I would have been completely lost in this project without him.

In addition, I would like to thank Henry Anderson for the hours upon hours he spent teaching me and helping me over the past year. He was the first person I went to with any technical questions, and he always took time to thoroughly explain anything I had questions on.

I would like to acknowledge the Jacobs' National Security group for the opportunity to be their first undergraduate researcher here at UTA. Their research questions and topics have played a huge role in steering my personal research and interests. They have given me a glimpse of the big picture of my field, and what issues companies are facing.

I would like to thank Dr. Laurel Stvan, Professor and Chair of the Department of Linguistics. She has been extremely accommodating and flexible with my endless list of

unorthodox requests for classes and scheduling. Without her going out of her way for me, I would not have been able to take the courses I needed to graduate this semester.

Last, but certainly not least, I would like to thank my husband, Cody. He has been there for me and helped me through every idea, frustration, and long night working on this project. He has been my personal programming tutor, voice of reason, emotional support, and much-needed shoulder masseuse.

April 20, 2018

ABSTRACT

IMPROVING AUTOMATIC SUMMARIZATION FOR LOW- AND MODERATE-RESOURCE, MORPHOLOGICALLY COMPLEX LANGUAGES

Kalen Goss Manshack, B.A. Linguistics

The University of Texas at Arlington, 2018

Faculty Mentor: Pete Smith

Resource-poor, morphologically complex languages are at a disadvantage in natural language processing tasks, such as automatic text summarization or machine translation, due to the shortage of quality linguistic data available in these languages. Recently, researchers have introduced a language-independent, centroid-based method for automatic text summarization which garnered international attention for its success. This thesis explores methods for improving Rossiello et al.'s summarization approach on resource-poor, morphologically complex languages by implementing additional preprocessing steps on the data. Thereafter, stemming is shown to marginally improve research benchmark ROUGE scores for summarizations in German, a relative morphologically complex language, as well as in Turkish, an agglutinative language. In

addition, a manual semantic analysis of the associated Word2Vec models in this approach showed improved accuracy when models were constructed on stemmed corpora. This result has implications for research on word embeddings in low-resource and morphologically complex languages.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Natural Language Processing, or NLP, sits firmly at the intersection of linguistics and computer science. It aims to computationally process and analyze human language, and is data-driven—meaning its methods rely on large amounts of linguistic data. The early 21st century is often called the "Information Era" due to the staggeringly large amount of information and data available to the average person; but these massive amounts of data are not distributed equally across the many languages of the world. Linguistic data is most often gathered from the internet—in the form of social media posts, literary data, or recordings of speech—and though English is not the most widely-spoken language on the planet, it does have the largest internet presence and the most linguistic data available. Therefore, natural language processing research is predominately conducted in the English language.

Most languages spoken around the globe do not have a large internet presence or significant amounts of linguistic data. These languages are called low-resource languages, referring to the amount of linguistic data available in that language. Because of this lack of data, researchers in these low-resource languages have difficulty implementing many methods of natural language processing. These difficulties are further exacerbated when a language is also morphologically complex, and therefore poses its own set of unique problems for NLP. Those languages that are both low-resource and morphologically complex fall significantly behind more resource-rich, morphologically simple languages

in terms of natural language processing advancements. As a result, NLP tools that are considered essential to many businesses—such as machine translation, speech recognition, chat bots, and sentiment analysis—often perform poorly in these languages. Therefore, low-resource, morphologically complex languages are in need of additional linguistic data, as well as research into a more efficient utilization of existing data.

The experiments described in this paper explore how different preprocessing methods affect NLP output in morphologically complex languages, and how those effects scale to extremely small linguistic datasets. This paper strives to answer the specific question, "Does stemming the training corpus of centroid-based summaries improve results in low-resource, morphologically complex languages?" The initial experiments were performed with automatic text summarization—a method of NLP—but the questions that arose from those results prompted further research into the quality of the language models used for summary production.

The results showed a considerable change in the summaries generated and highlighted the difficulties in objective summary evaluation. A qualitative analysis of the language models used to create the summaries suggests that additional preprocessing steps can improve the quality of language models trained with small, morphologically complex corpora.

CHAPTER 2

LITERATURE REVIEW

2.1 Automatic Text Summarization

The idea of automatic text summarization first garnered academic attention in the 1950s, with Hans Peter Luhn's *The Automatic Creation of Literature Abstracts* (1958). In his paper, Luhn used an IBM 704 data-processing machine to scan scientific documents and choose sentences from the document to make up its abstract, or summarization. The sentences were extracted individually from the document and each assigned a *weight*, or score, which was determined by word and phrase frequency. Sentences that contained words with a higher frequency in the document were given higher weights—ignoring *stop words*, which are common, high-frequency words, such as "and," "the," and "but." Though technology has changed significantly since the IBM 704, this conceptual approach to automatic text summarization has not.

2.1.1 Extractive Summarization

Though there are a variety of automatic text summarizers in use today, there are two main schools of thought for the underlying theory: abstractive and extractive. *Extractive text summarization* uses existing sentences from the document to create the summary, whereas *abstractive text summarization* involves creating entirely new sentences for the summarization, and is currently more theory than practice. Abstractive summarization is much more challenging to implement; and for that reason, most research

to date has been focused on extractive summarization. As of yet, there are no fully abstractive summarizers in existence (Allahyari, et al., 2017).

Since Luhn's summarizer in 1958, a multitude of other extractive summarizers have been created. Most of these follow Luhn's general concept: assign a score or weight to each sentence in a document and create the summary from the *n* highest weighted sentences. The differences in approach are found in *how* the sentences are weighted.

2.1.1.1 Topic Representation Approaches

In 1993, Ted Dunning introduced a method expanding upon Luhn's in which the log-likelihood ratio, instead of raw frequency, was used to determine the *topic signature*—the words within the article that best describe the topic. In his approach, he used the log-likelihood ratio test to find *bigrams* (groupings of two words) that appeared together with much higher frequency than would be expected normally and added them to the topic signature. His method was very effective and has since been widely cited, especially in the news domain (Harabagiu & Lacatusu, 2005).

The two most popular weighting techniques in extractive summarization are word probability and TFIDF. *Word probability* is the simplest frequency-driven summarization method. Much like Luhn's approach, word probability uses the frequency of words to decide which are the most important to the document. A commonly cited summarizer using this simplistic method is SumBasic (Nenkova & Vanderwende, 2005), which assigns sentence weights based on the average probability of the words in the sentence. The *probability* of a word is determined using a corpus, where it is equal to the frequency of the word in the corpus divided by the total number of words in the corpus.

A significant drawback to probability models is that they rely on a list for stop word filtering. An alternative to this method which does not need a stop word list is *TFIDF,* which stands for *Term Frequency ∗ Inverse Document Frequency*. This method is widely used in summarization, as well as other natural language processing tasks. It assesses the weight of a word by using the equation:

$$word\ frequency\ in\ the\ document\ \times \log\left(\frac{word\ frequency\ in\ the\ corpus}{documents\ in\ the\ corpus\ in\ which\ the\ word\ appears}\right)$$

An effective approach that uses TFIDF is *Latent Semantic Analysis* (LSA), which was introduced by Deerwester et al. (1990). This method builds a term-sentence matrix in which each row is a word and each column corresponds to a sentence. LSA allows for multiple topics to appear within a single document. Gambhir & Gupta (2017) published a comprehensive survey of modern automatic summarization techniques.

2.1.2 Summary Evaluation

Objectively evaluating the quality of a summary is a very difficult task because there is no objective right or wrong summary for a given text. Human-made summaries are often valued as the gold standard for summaries, but it has been found that humans themselves are very inconsistent in both the summaries they produce and their assessment of other summaries. There are multiple automatic summarization metrics used by researchers, but none as of yet can accurately recognize a quality summary (Allahyari, et al., 2017).

2.1.2.1 Human Evaluation

The original and simplest method of summary evaluation is human judgement. In this method, humans are given the automatically generated summaries to score. In some cases, they are asked to choose which summary is better among a selection; in other cases

they are given a rubric to use in evaluation of the summaries, which can include topics such as redundancy, topic coverage, grammaticality, and readability. The yearly Text Analytics Conference (TAC) held by the National Institute of Standards and Technology (NIST) has used different human evaluation methods from year to year. In 2008, they used a method in which the summary assessor was given a list of questions on the key points of the document and was tasked with answering them using only the generated summary. The more answers to the key questions that a summary had, the higher its evaluation score (TAC 2008 Opinion Summarization Task Guidelines, 2008).

In 2003, Radev and Tam proposed a structure to human evaluation called *relative utility*, in which multiple judges are given a single sentence at a time from the document and asked to rate it on a scale of 1 to 10 in terms of its suitability for the summary. The average score from all the judges is then attached to each sentence, and summaries are given the points associated with each of their extracted sentences and evaluated on the total number of points earned. However, it is very tedious and time consuming to have multiple judges individually score each sentence, especially when given a long document or even multiple documents (Lloret, Plaza, & Aker, 2017).

2.1.2.2 ROUGE

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is the most widely adopted metric in automatic summary evaluation, despite its criticisms (Sjöbergh, 2007). ROUGE generates three word-level evaluations comparing the automatically generated summary with a human-made gold standard summary. The metrics are *recall*, which is the percentage of words in the automatically generated summary that also appear in the human summary, *precision*, which is the percentage of words from the human summary that also

appear in the automatically generated summary, and the *F-Score*, which is the weighted average of the two. The scores are output as decimals between 0 and 1 but are most often reported as percentages on a 0-100 scale. In cases where there is a constraint on the summary size, the recall score alone is usually reported. In his paper introducing ROUGE, Lin (2004) tested his method on the datasets from the 2001-2003 Document Understanding Conferences (DUC), and produced summary scores that were 70% similar to the human evaluations (the correlation reached 90% when the summary was allowed to be very long). Since then, ROUGE has been used in many of the field's most prestigious conferences, such as DUC and TAC. Since its debut at the 2004 DUC, ROUGE has been praised for its accuracy as a monetarily, computationally, and logistically low-cost evaluation package (Lloret et al., 2017).

ROUGE's biggest criticism is that it only assesses strings (which are finite segments of characters, such as letters or numbers) and does not take into account meanings expressed by these words or phrases. "The phrase 'large car' in a system summary, for example, would not match 'large green car' in the gold standard summary, despite 'large' and 'green' independently modifying 'car'" (Tratz & Hovy, 2008, p. 1). Sjöbergh (2007) strengthened these criticisms by using a simple summarization method with a greedy word selection strategy (an algorithm that chooses the locally optimal choice at each pass as a path to find the global optimum). With this method, he showed that his summarizer could generate summaries with high ROUGE scores that were objectively poor when assessed by human judges. He was able to do this by choosing the more frequent bigrams in the document; while these were ungrammatical and nonsensical when strung together, they produced very high ROUGE scores because they had many of the same words that were

```
of the hurricane andrew had been injured and the storm caused
by the gulf of mexico and louisiana and at least 150 000 people
died on the us insurers expect to the florida and there are
likely to be concentrated among other insurers have been badly
damaged a result of damage caused in the state and to dollars
```

Figure 2.1: A Summary Created by Sjöbergh's Summarizer (Lloret et al., 2017, p. 8)

used in the human-made summary. Figure 2.1 shows a section of one of the summaries created by Sjöbergh's summarizer, whose summaries scored upwards of 41%, which is considered state-of-the-art in natural language processing.

Another common criticism of the ROUGE metric is that it can only consider one gold standard summary at a time, causing the results to be subjective. If the automatic summary contains objectively good summarization sentences that are not included in the human summary, ROUGE does not in count in favour of the automatic summary (Lloret at al., 2017).

There have been many evaluation metrics created to address ROUGE's drawbacks, though none have gained the same popularity. One such metric is ROUGE-C, which was created by He et al. (2008) to address the drawbacks of ROUGE needing a gold standard summary for evaluation. ROUGE-C instead uses the original article to evaluate the summary. This metric is not entirely independent, however, as it requires human-provided "query-focused information" (Lloret, Plaza, & Aker, 2017, p. 9). Still, it has proven to be far less time consuming than traditional human methods, and has also proven to correlate well with human-made summaries (He, et al., 2008). There are many other evaluation metrics in use, such as DEPEVAL (Owczarzak, 2009), FRESA (FRESA 2.1 FRamework for Evaluationg Summaries Automatically, n.d.), and ROUGE-WE (Ng & Abrecht, 2015), which will be discussed in a later section.

2.2 Word2Vec

The *distributional hypothesis* was first introduced by J.R. Firth in 1957 and states that words found in the same contexts throughout language have similar meanings. This is the driving theory behind *vector space models*, which are algebraic models used to represent words in a continuous vector space. Words are mapped into the vector space according to the contexts in which they are found. Therefore, words that appear in similar contexts are mapped to similar vector representations, and are thus closer together in the vector space. The term *word embedding* refers to these vector representations of words. Figure 2.2 shows a 3D representation of word embeddings, with each point in the space representing a word.

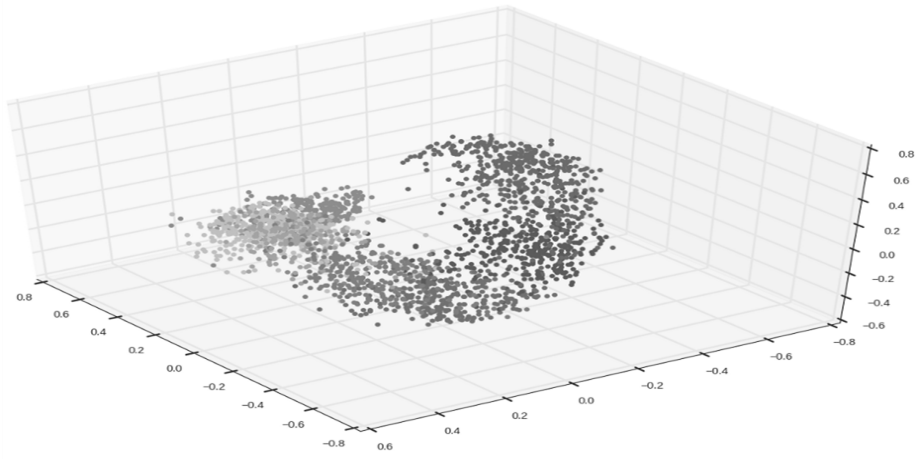

Figure 2.2: A 3D Representation of Word Embeddings (Boukkouri, 2017)

These word embedding models can be seen as semantic "maps" of a language. It has been found that certain directions in the vector space correspond with certain semantic relationships, such as gender or verb tense. Figure 2.3 shows a simplified visualization of this phenomenon.

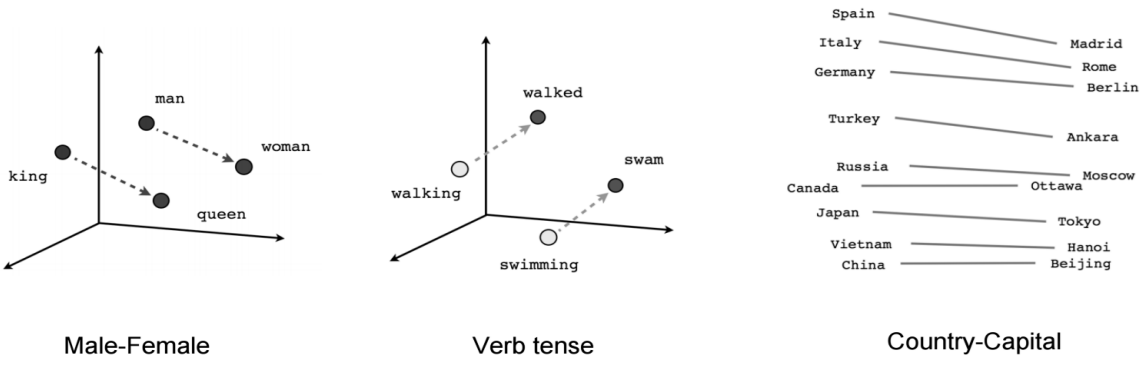

Figure 2.3: Semantic Relationships Learned Within Word Embeddings (Vector Representations of Words, 2018)

One of the most popular word embedding algorithms used today is Word2Vec, introduced by Tomas Mikolov (2013a). The models available in Word2Vec are CBOW and skip-gram. *CBOW* stands for Continuous Bag of Words, and refers to a method which predicts each word using its surrounding words, without taking into account word order. Like the name suggests, it looks at the surrounding words as if they were randomly thrown into a "bag." In the continuous *skip-gram* model, on the other hand, each word is used to predict its surrounding words, and therefore word order plays an important role. Generally, the CBOW model has proven to be much faster while the skip-gram model, though slower, does a better job when confronted with infrequent or out-of-vocabulary words (OOVs, which are words not in the Word2Vec model) (Mikolov, Chen, Corrado, & Dean, 2013b).

2.2.1 Applications

Word embeddings have gained massive popularity in natural language processing since the introduction of Word2Vec. Today, they are used in companies around the world for tasks such as sentiment analysis, machine translation, and recommender engines. One of the most impactful and promising applications of word embeddings is in word-level machine translation. Researchers have found that word embeddings in different languages

build into surprisingly similar shapes. The lexical item "dog" in a word embedding model trained on English has a very similar vector representation to its Spanish equivalent, "*perro,*" in a model trained on Spanish (Zou, Socher, Cer, & Manning, 2013).

Embeddings are not only being used in research. Clothing company StitchFix uses word embeddings to find clothing specific to users' tastes, as well as to summarize and analyze the sentiment of reviews for their clothing. They even use embeddings to keep track of users' pregnancies in order to send them clothes that will fit them correctly in each term (Moody, 2015). The music streaming giant Spotify also uses word embeddings for their music recommendation system (Kumar & Samuels, 2015).

Word embeddings have given insight into the effects of societal expectations on language. Word embeddings draw conclusions and parallels from language that many native speakers do not consciously notice. For example, Kheyrollahi (2015) trained a word embedding model on a wide range of domains, such as politics, sports, arts, culture, and technology; and when presented with "president" – "power," the model returned "prime minister." While this relation is accurate, this is not often expressed explicitly in language. Other conclusions the model drew were "Iraq" – "violence" = "Jordan," "library" – "books" = "hall," and "human" – "animal" = "ethics." However, there are also some downsides to embeddings reflecting the human psyche. Word embeddings have been shown to reflect—or even amplify—societal stereotypes, especially in terms of gender. In 2016, Bolukbasi et al. published a paper titled *Man is to Computer Programmer as Woman is to Homemaker? Debasing Word Embeddings*, in which they dive into *how* embeddings are coming to these biases and, worse, how they are actually amplifying them.

Word embeddings are also being used for sentiment analysis in applications such as restaurant and product reviews. Hamilton et al. (2016) used word embeddings to analyze how the same word can have an entirely different sentiment in different settings. They studied the use of the word “soft” in the online platform Reddit; specifically, they studied its negative or positive connotation on the pages r/Sports and r/MyLittlePony. Their findings can be found in Figure 2.4, which shows that “soft” had an extremely positive sentiment in r/MyLittlePony, and an extremely negative sentiment in r/Sports, illustrating some of the larger issues facing sentiment analysis.

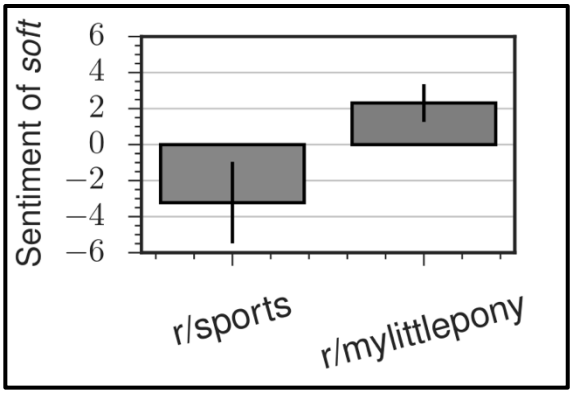

Figure 2.4: Sentiment of "Soft" in r/Sports and r/MyLittlePony (Hamilton et al., 2016)

Word embeddings have also been used to track the sentiment of words over time, as in Figure 2.5.

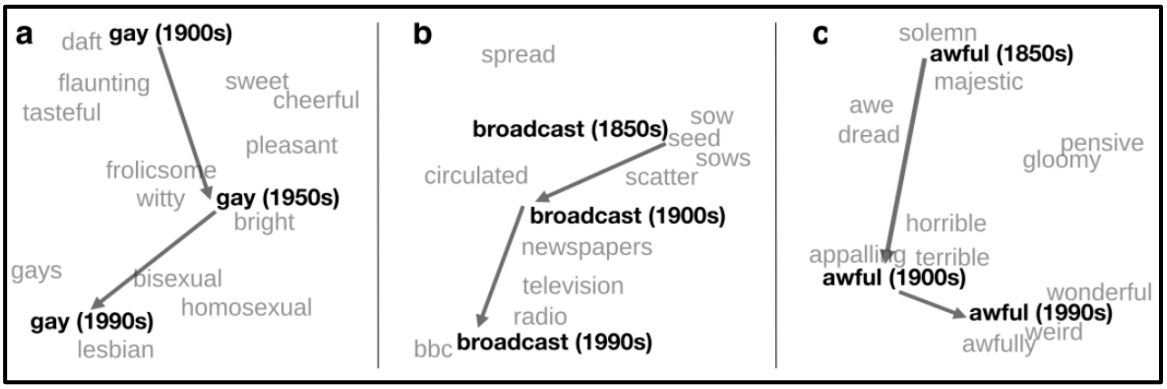

Figure 2.5: Word Sentiment Over Time (Ruder, 2018)

2.2.2 Centroid-Based Summarization

In physics and mathematics, the centroid of a shape or set of points is the mean of all points in that shape or set. In geometric terms, it is the point on a shape where, when placed upon the tip of a pencil, the shape would be perfectly balanced. In the context of word embeddings, the centroid is the average of all the word vectors that make up the embeddings. In a geometric model of the vector space, it would be the center-most point. In centroid-based summarization, the *centroid embedding* is the center of the *theme* (or themes) of the document (Radev, Jing, & Budzikowska, 2004).

Centroid-based summarization is a concept that was introduced by Radev et al. in 2004. First, words of the document are ranked by their TFIDF weights, and then the *n* highest-ranked words are selected to make up a mini document. The centroid is the average of all the word vectors that make up the mini document. For summarization, the word vectors of each sentence in the document are averaged to create sentence embedding points. The sentence embeddings that are closest to the centroid are chosen for the summarization, with the sentence or word limit being decided by the user (Radev et al., 2004). Figure 2.6 shows a 2D representation of the centroid vector, marked as “centroid,” and the sentences of the document, each marked with numbers. The bolded numbers close to the centroid mark are the sentences that were chosen for the summarization.

Though centroid-based summarization has been proven to perform well when assessed by human judges, it does not fare as well when assessed by ROUGE (Wong, Wu, & Li, 2008), as embeddings represent semantic meaning, and ROUGE only assess

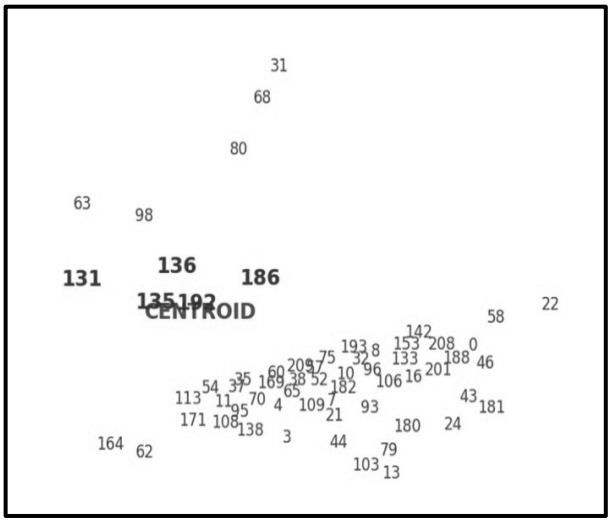

Figure 2.6: 2D Visualization of Centroid-Driven Summarization (Rossiello, Basile, & Sameraro, 2017)

orthographic words. In terms of a previous example, the vector representations of "large car" and "big sedan" would be very similar, where ROUGE would assign them 0 correlation.

2.2.2.1 ROUGE-WE

One of the metrics created in response to ROUGE is ROUGE-WE (Ng et al., 2015). In this metric, summaries are evaluated using pre-trained word embeddings from a Word2Vec implementation. The words of the summaries are scored based on their semantic similarity to the words in the gold standard. This metric has shown promising results by matching human-made judgements with up to 91% correlation. However, the implementation is still currently reliant on a pre-trained English word embedding model with no option for the user to provide their own model. Therefore, there is still no support for other languages, though Ng and Abrecht have expressed intentions to update the code in order to allow a model to be loaded locally (Ng et al., 2015).

2.3 Resource-Poor Languages

There are approximately 7,000 languages in the world today, but only 20 languages that are considered high-resource, or resource-rich (Duong, 2017). *Resource-rich* languages are categorized as languages with large amounts of linguistic data available, such as English, the most resource-rich language in the world. Most of these 7,000 languages are *resource-poor*, meaning they lack "not only practical NLP systems, but even the large labeled corpora typically used to develop such systems" (Baumann & Pierrehumbert, 2014).

2.3.1 Need for Natural Language Processing

As the world becomes more interconnected and globalized, the need for language data and natural language processing systems is increasing rapidly, and the gap in NLP resources between resource-rich and resource-poor languages is growing. For example, countries that are part of the European Union and other international establishments receive the benefit of large amounts of linguistic data through multilingual communication, such as legislation. These *parallel texts*, or documents with identical content in different languages, are an essential tool for building machine translation engines. Countries that are not participants in these types of establishments have far fewer opportunities to collect linguistic data (Nakov & Ng, 2012).

According to Oracle Corporation, *structured data*—that which is organized and often parsed or tagged (including parallel texts)—only accounts for about 20% of the generated data in the world. The rest is scattered and buried in social media, e-mails, articles, blogs, news websites, and anywhere else there is digital language. Natural language processing handles this unstructured, messy language data, and is used to for

internet search results, election polls and approval ratings, machine translation for international communication, question answering for customer service chat bots, filtering spam e-mail, automatic summarization, and many more applications. Languages need these opportunities as part of developing and taking part in the globalization of the Information Era. Figures 2.7 and 2.8 show this large disparity in the languages used on the internet. Though the internet is not the only source of linguistic data, it is by far the largest.

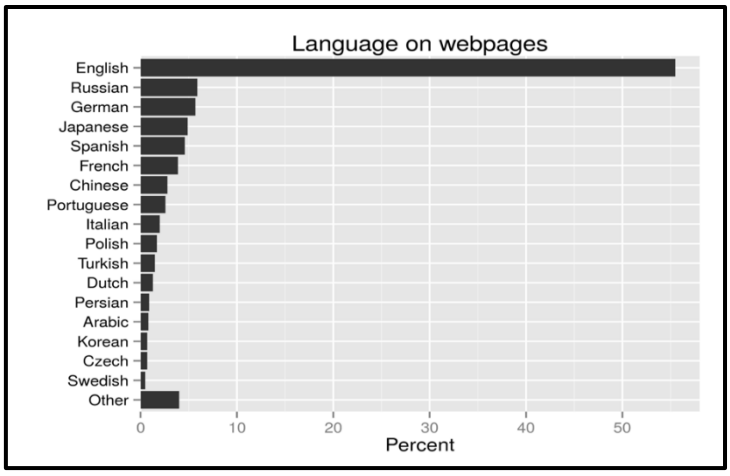

Figure 2.7: The Top 17 Languages Used on the Internet (Plottingman, 2015)

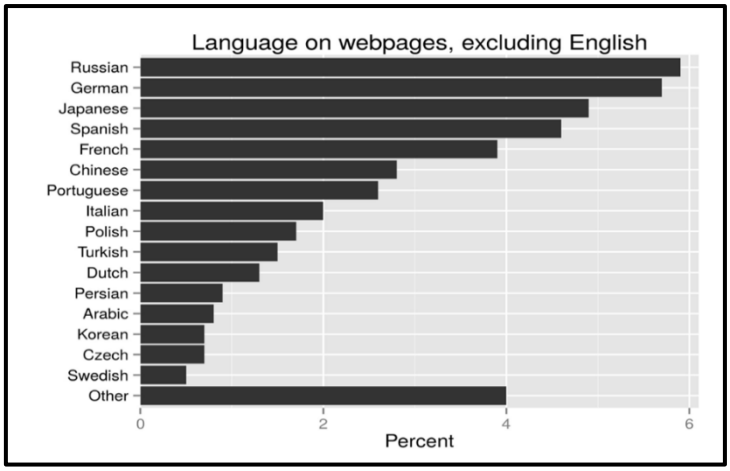

Figure 2.8: The Top 17 Languages Used on the Internet, Excluding English (Plottingman, 2015)

One of the largest obstacles facing resource-poor languages in terms of data collection is the percent of internet users. The more speakers of a language that use the

internet, the more language data there is from their blogs, webpages, social media, and e-mails. Therefore, internet user percentage has a huge impact on ease of data collection. Though 47% of the world's population has reported using the internet in the past year (International Telecommunications Union, 2017), these users are very unevenly distributed. While an average of 72% of Europeans and North Americans are internet users, that number falls to just 25% in Africa. India, the second most populated country in the world, reports only 29% of its citizens use the internet. Indonesia, the ninth most populated country in the world, is 157th in internet user percentage, with only 25%. 36 out of the 100 most populated countries in the world report internet user percentage below 50%, and 12 of them report internet user percentage below 25% (International Telecommunications Union, 2017).

2.3.2 Interest in the United States

The need for natural language processing in these low-resource languages extends beyond speakers of the language, or countries who claim it. International relations rely on shared language understanding; and governmental agencies and corporations around the world are researching and using natural language processing in resource-poor languages. In 2017, the CIA announced that it is actively hiring for positions in over 60 resource-poor languages that are eligible for their Foreign Language Program (The Central Intelligence Agency, 2017); these include Burmese, Indonesian, Tagalog, Tibetan, Urdu, Pushto, and Zulu. The U.S. Government is also offering scholarships for students wanting to study abroad to learn Azerbaijani, Bangla, Hindi, Indonesian, Korean, Punjabi, Swahili, Turkish, and Urdu (Bureau of Educational and Cultural Affairs Exchange Programs, n.d.); and the United States Defense Intelligence Agency has "immediate" openings in Pashtu, Somali,

Urdu, and more (The Defense Intelligence Agency, n.d.). The need for speakers and researchers in these languages arises from the unique position of the United States Government. Though the United States is extremely active on the world stage, with troops stationed in nearly 150 countries (CNN, 2011), it also has one of the lowest percentage of bilingual speakers in the developed world, with only about 15% of U.S. Citizens knowing more than one language proficiently. In comparison, 56% of Europeans are multilingual. Mahmoud Al-Batal, Arabic professor at the University of Texas at Austin, describes the issue by saying that the inability to speak a foreign language makes it difficult for Americans to compete globally on a linguistic and cultural level (Franklin, 2013). Because of the shortage of bilingual speakers in these desired languages, the U.S. Government, and companies that do business abroad, are in need of natural language processing resources and tools for these low-resource languages, especially in the field of machine translation.

2.3.3 Research in NLP in Resource-Poor Languages

Researchers have attempted to combat these stifling odds through creative data-collection practices. Researcher Sean Packham (2016) proposed a method to use internet crowdsourcing (enlisting the help of the masses) to create parallel corpora. In his paper, Packham highlights the issues facing researchers of the South African language isiXhosa. He states, "Researchers have been unable to assemble isiXhosa corpora of sufficient size and quality to produce working machine translation systems and it has been acknowledged that there is little to no training data" (Packham, 2016, p. iv). Packham proposes an internet game in which native speakers of isiXhosa will be inadvertently providing English translations of isiXhosa text to create a parallel corpus. Another method for collecting data for low-resource languages is using social media. In their paper *Leveraging Twitter for*

Low-Resource Conversational Speech Language Modeling, Jaech and Ostendorf (2015) propose a method of using Tweets from native speakers of four low-resource languages to collect language data.

In addition to these efforts in data collection, some researchers, such as Rhoit Dholakia (2014), are also studying ways to use already-collected data in a more efficient and effective manner. Dholakia is part of a research movement that is studying the use of a *pivot language*—a third, intermediary language—to translate from one language to another when there are not sufficient parallel corpora between the original languages.

2.4 Morphologically Complex Languages

Natural language processing tasks—even the training of word embeddings—use *words*, which, in computer science, are groupings of characters, or letters separated by spaces (in most writing systems). Many NLP methods rely heavily on the assumption that each meaning is connected to a word and each word is connected to a meaning (Nakov & Ng, 2011). While no language perfectly conforms to these expectations, some flout them more than others.

Morphology refers to "the component of mental grammar that deals with types of words and how words are formed out of smaller meaningful pieces and other words," (Department of Linguistics at Ohio State University, 2011, p. 148). This means that the morphology of a language refers to the individual units of meaning—or *morphemes*—that constitute a language. To give an example in English, the word "unstoppable" is made up of three separate morphemes: *un*, meaning "not," *stop*, meaning "to cease, or cause to cease," and *able*, meaning "capable of." These three morphemes and their meanings come together to create a word that means "is not able to be stopped." Most words in English can

be broken up in to these morphemes, and many words are single morphemes themselves, such as “free,” “cat,” or even “piano.”

2.4.1 The Spectrum of Morphological Complexity

A *morphologically complex* (or *morphologically rich*) *language* is “a language in which grammatical information is expressed at word-level through affixes” (Tsarfaty, et al., 2010, p. 1). In contrast, *morphologically simple* languages rely on word order and individual, separate words to relay the same information. Though all languages are made up of morphemes, there is a spectrum on which a language can lie in terms of its morphological complexity. On one end of the spectrum are the morphologically simple languages such as Mandarin Chinese. Further down the spectrum—though quite a bit closer to morphological simplicity than complexity—is English. Closer to the morphologically

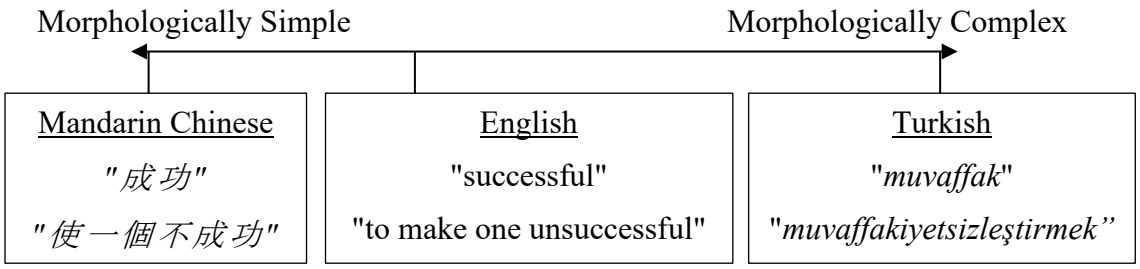

Figure 2.9: Spectrum of Morphological Complexity

complex end of the spectrum is German, whose characteristically long words are often made up of multiple words and morphemes. At the very end of the spectrum are the extremely morphologically complex languages, such as Turkish and Hungarian. Figure 2.9 shows an example demonstrating this spectrum of morphological complexity.

2.4.2 Difficulties for NLP

Because of their word-level intricacies, morphologically complex languages pose a unique challenge for natural language processing. Two common steps for preprocessing

in NLP are stemming and lemmatizing. *Stemming* involves removing all suffixes from a word, while *lemmatization* converts a word to its root form. The key difference between the two approaches is that stemming, while quicker and easier to implement, simply "chops off" the end of the word, with the goal of "reducing all words with the same root… to a common form" (Lovins, 1968), whereas lemmatization uses a pre-built, human-made knowledge base to convert words to their root form (Manning, Raghavan, & Schütze, 2008).

These methods most often perform sufficiently well in morphologically simpler languages, but have severe difficulties in morphologically complex languages. The issue arises from the fact that affixes in morphologically complex languages relay grammatical information that is key to the meaning of the utterance. For example, the word *durdurulamaz* in Turkish means "one who is unstoppable." If the word were traditionally stemmed, and all suffixes were removed but the root, the word would simply be, *dur*, meaning "stop." On the other hand, if every word of a Turkish corpus is left as-is, many words of the vocabulary would only appear once, as there are a near unlimited number of morpheme combinations that can make up a word. In this case, an NLP system would have a very hard time trying to process any previously unencountered combination of morphemes, even though it had encountered each morpheme encapsulated in other words. An extremely large Turkish corpus would be needed to perform any basic natural language processing task effectively. This is no small feat, given that Turkish—as well as most morphology complex languages—is a relatively low-resource language.

Natural language processing in morphologically complex languages is an active and necessary area in research. In 2003, Koehn and Knight proposed a method to

automatically split German compound words into their smaller, single-word parts. Though their method had only a small effect on the BLEU scores (a metric to measure machine translation results) of their test sentences, it was proven to accurately split many German compound words. Some researchers, such as Nakov and Ng (2011), are approaching the issue by using morphologically similar words and phrases to develop a paraphrasing technique for machine translation. Others are using the pivot language method mentioned above (Kholy, 2016).

2.4.2.1 Modern Stemmers

Importantly, researchers are also developing more comprehensive stemming and lemmatizing algorithms for these morphologically complex languages (most of which did not have stemmers until recently), such as Osman Tunçelli's Turkish Stemmer (2015), which claims to account for many of the issues facing Turkish NLP, such as vowel harmony and affix separation. The most commonly cited weakness of stemmers is that they are too aggressive (they remove too much), but newer stemmers are addressing these issues by using more complex and meticulous stemming algorithms. In 2017, Weißweiler and Fraser tested the results of existing German stemmers while also proposing their own, which performed significantly better than previous stemmers in both recall and F-score (it was barely out-performed in precision by the UniNE Light stemmer). Figure 2.10 shows a visual from their paper in which the stemmers they were testing (including their own, CISTEM), were fed the German words *Adler, Adlers, Adlern,* and *adle.* The first three are the German word for "eagle," with different case endings, and the last is a form of the verb

"to enoble." Only CISTEM was able to distinguish the words for "eagle" from the verb "to enoble."

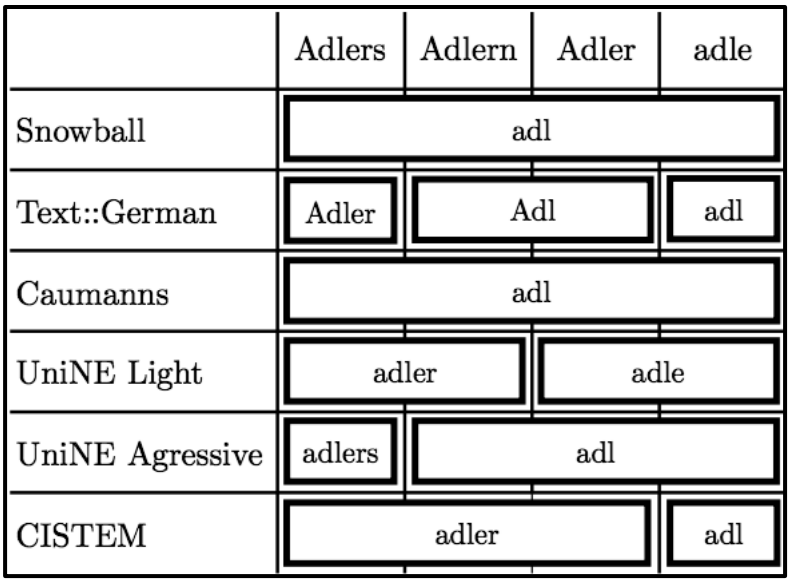

Figure 2.10: CISTEM Results (Weißweiler & Fraser, 2017)

2.5 Centroid-Based Text Summarization Through Compositionality of Word Embeddings

This thesis was inspired by the work of Rossiello et al. in their 2017 paper "Centroid-Based Text Summarization Through Compositionality of Word Embeddings," which was presented at the MultiLing 2017 Workshop on Summarization and Summary Evaluation Across Source Types and Genres. In the paper, Rossiello et al. proposed a "centroid-based method for extractive summarization which exploits the compositional capability of word embeddings" (pg. 20), which incorporated methods (such as skip-gram) from Tomas Mikolov's (2013) paper "Distributed Representations of Words and Phrases and their Compositionality" into Radev et al.'s centroid summarization method (which uses only CBOW for embedding training, and therefore does not take word order into account). Mikolov's methods introduced context and word order to the centroid summarization's embedding process.

For training data, Rossiello et al. used Wikipedia database dumps, which are freely available backups of every Wikipedia article for a given language. For languages that have their own Wikipedia, these dumps provide an invaluable source of linguistic data. Word embedding models were then trained on these database dumps in English, Italian, German, Spanish, and French. For preprocessing, the texts were converted entirely into lower-case, tokenized into words (*tokenization* is the process of splitting raw text data into individual pieces for NLP tasks), and all stop words were removed. No stemming was performed, with the expectation that the embeddings would learn the necessary linguistic connections between words with the same root. Figure 2.10 shows the ROUGE-1 and ROUGE-2 recall scores achieved by the researchers (listed as "C_W2V") compared with common summarization baselines. *ROUGE-1* measures the overlap in single words, and *ROUGE-2* measures the overlap of bigrams. More details of Rossiello et al.'s methods and results are discussed in the chapters ahead.

	English		Italian		German		Spanish		French	
	R1	R2	R1	R2	R1	R2	R1	R2	R1	R2
LEAD	44.33	11.68	30.46	4.38	29.13	3.21	43.02	9.17	42.73	8.07
WORST	37.17	9.93	39.68	10.01	33.02	4.88	45.20	13.04	46.68	12.96
BEST	50.38	15.10	43.87	12.50	40.58	8.80	53.23	17.86	51.39	15.38
C_BOW	49.06	13.43	33.44	4.82	35.28	4.93	48.38	12.88	46.13	10.45
C_W2V	**50.43**‡	13.34†	**35.12**	**6.81**	**35.38**†	**5.39**†	**49.25**†	**12.99**	**47.82**†	**12.15**
ORACLE	61.91	22.42	53.31	17.51	54.34	13.32	62.55	22.36	58.68	17.18

Figure 2.11: ROUGE Results Reported from Rossiello et al.'s Summarizer (Rossiello et al., 2017)

CHAPTER 3

THE EXPERIMENT

The experiments performed in this thesis address the issues facing natural language processing—specifically automatic text summarization—in resource-poor, morphologically complex languages. Rossiello et al.'s centroid-based summarizer was used to test the effect that stemming the training data has on summarization results. Due to the issues in stemming morphologically complex languages, researchers have generally avoided stemming when working with these languages. As previously explained, word embeddings are mapped based on their context; therefore, words need to be observed in as many different contexts as possible for accurate and useful mappings to be made of them. Morphologically complex languages can have a near infinite combination of morphemes, and therefore a near infinite number of unique words with far less reoccurrences of each (compared to less morphologically complex languages). For this reason, word embedding methods traditionally do not perform well on morphologically complex languages.

Stemming significantly lowers the number of unique vocabulary tokens in a corpus while increasing the occurrences of the remaining tokens. It can be visualized as a cloud, with each word embedding being a water droplet, and the "strength" of the vocabulary referring to the number of occurrences of each token in the corpus. When there is a large amount of training data (a higher number of water droplets), the cloud is spread out over a large space while also remaining dense. This is akin to having an embedding model with a large, strong vocabulary. However, when the language is resource-poor, there are far less

water droplets. The cloud in which the training data is not stemmed is widely spread out and sparse. It has many different vocabulary tokens without sufficient instances of each, making them weak. Stemming the training data condenses the same cloud, with the same amount of water, into a smaller and more dense space—the vocabulary shrinks in number, but improves in strength. The question being tested here is whether denser embeddings of word stems are more effective for this summarization method than sparser embeddings on the full vocabulary.

By the end of experimentation, four separate experiments had been conducted. The first was a recreation of Rossiello et. al.'s Experiment, which aimed to recreate their original findings with the German language. The second experiment was the Summarization Experiment, which tested the quality of summarization outputs from Word2Vec models trained on stemmed and unstemmed corpora. The third experiment, the Turkish Summarization Experiment, was performed similarly to the second, but with Turkish data in place of German. The last experiment was not in the initial plan for this paper, and was created in response to the results of the second experiment. In this fourth experiment, the Word2Vec Model Accuracy Experiment, a semantic qualitative analysis was performed on each of the Word2Vec models from the second experiment to determine their semantic accuracy.

The second and fourth experiments were performed with the same German dataset as the first. German is neither resource-poor nor extremely morphologically complex. However, it is moderately morphologically complex; and because it is resource-rich, experiments can be done on differing sizes of training data that could not be done with a truly resource-poor language. German has been used as a "guinea pig" language for many

experiments in morphological complexity; as one researcher put it, “German has become almost an archetype of the problems caused by MRLs [Morphologically Rich Languages]” (Tsarfaty, et al., 2010, p. 1).

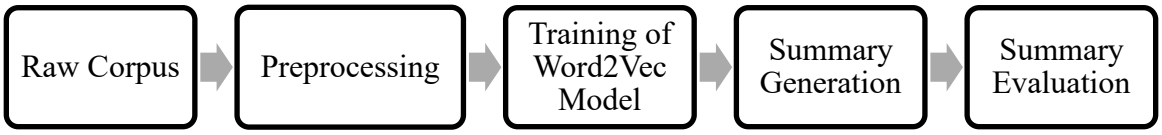

Figure 3.1: Illustrated Work Flow for Summarization Experiments

3.1 Methodology

3.1.1 Rossiello et al.’s Experiment Recreation

In order to validate future results, the first experiment aimed to replicate the results from Rossiello et al.’s experiment in German (in which they reported ROUGE-1 recall scores of 35.38%). Thus, each step of this experiment mirrored the methodology of Rossiello et al.’s experiment.

The German Wikipedia database dump from November 3rd, 2016, was used here, as well as in all German experiments in this paper. The raw Wikipedia dump was cleaned with the program WikiExtractor (Attardi, 2017), which removed the html markup from the files. Tokenizing was done using the popular Python library NLTK, stop words and punctuation were removed, and each word was lower-cased. Stemming was not performed.

The dataset was then used to train a Word2Vec model which used hierarchical and negative sampling, an 8-word symmetric window, and with 5 iterations over the corpus. The summarizations were made using the Python code from Rossiello et al.’s centroid-

based summarizer.[1] For each document, the summarizer created a centroid containing all words from the article with a TFIDF weight greater than 0.3. Then a summary was created from those sentences closest to the centroid until the word limit had been reached. Summarizations were made on the MultiLing 2015 testing dataset. This dataset consists of 30 articles and their gold standard human-made summaries (one for each article) in 38 languages, including German, Indonesian, and Turkish. In addition to the articles and their summaries is a suggested character limit for each summary.

In Rossiello et al.'s experiment, hyperparameter optimization of the summarizer was re-implemented with each individual article, and the word limit was changed per-article to reflect the suggested character limit supplied with the gold standard summaries. However, these labor-intensive measures were infeasible for this experiment due to computational limitations, time constraints, and the sheer number of possible parameters between the summarizer and ROUGE metric. Instead, the default parameters of Rossiello et al.'s summarizer were used, and the word limit was chosen by adding the average word count of all gold standard summaries to half of the average word count per sentence. For German summaries, the average word count per summary was 143, and half of the average word count per sentence was 7 (half of 14), therefore the summary word limit was set to 150.

Rossiello et al.'s experiment used ROUGE version 1.5.7, which is a version of ROUGE that is no longer available. The only currently available version of ROUGE is 2.1.2. Though some of the parameters available in ROUGE 1.5.7 are no longer offered in

[1] https://github.com/gaetangate/text-summarizer

version 2.1.2, the available parameters were set to best match those in the Rossiello et al. experiment. Stop words were not removed in the analysis of the summaries, and words with the same stems were considered matches. Despite the lack of hyperparameter optimization, article-to-article word limit changes, and available ROUGE parameters, the summarizer was still able to obtain an average recall of 24.71%, with some summaries reaching up to 30.0%.

3.1.2 Summarization Experiment

3.1.2.1 Preprocessing

The goal of the Summarization Experiment was to test whether stemming the Word2Vec training data improved the summarization results of Rossiello et al.'s summarizer. In this experiment, the same Wikipedia dataset dump and WikiExtractor were used to prepare the training data. Two separate preprocessed corpora were made from the Wikipedia corpus. The first was created using the preprocessing steps from Rossiello et al.'s Experiment Recreation, and was used to create the baseline summaries. In the second, the additional step of stemming was added to preprocessing; and this corpus was used to create the test summaries. The stemmer used in these experiments was the Snowball German stemmer, a popular and simplistic German stemmer offered in the NLTK package.

3.1.2.2 Model Creation

Once the two preprocessing corpora were made, six separate Word2Vec models were trained: three unstemmed baseline models and three stemmed test models. These models were trained using hierarchical and negative sampling, an 8-word symmetric window, and with a minimum word count of 5. Instead of implementing 5 iterations over the corpus for every model—as in Rossiello et al.'s original experiment—that number was

reduced to 1 for the sake of time and computational efficiency. The minimum word count was lowered from 10 to 5 due to the small size of the corpora. The three baseline models were trained on the full training corpus, 10% of the training corpus, and 1% of the training corpus, respectively. The three test models were trained on the full stemmed training corpus, 10% of the stemmed training corpus, and 1% of the stemmed training corpus, respectively. The stemmed and unstemmed models of each size were trained on identical portions of the corpus (before stemming). These model sizes were chosen to reflect Wikipedia sizes in low-resource languages; the Turkish and Indonesian Wikipedias are each approximately 10% the size of the German Wikipedia, and the Tagalog Wikipedia is approximately 1% the size. Table 3.1 shows the number of unique vocabulary tokens in each model. The last column shows the percent decrease in tokens from the unstemmed model to the stemmed model.

Table 3.1: Vocabulary Sizes of Models

Name of Model	# Vocab Tokens	% decrease
Full German	2,079,855	18%
Full Stemmed German	1,692,738	
10% German	435,831	18%
10% Stemmed German	357,015	
1% German	82,319	17%
1% Stemmed German	68,664	

3.1.2.3 Summarization and Evaluation

Summarizations were made on the MultiLing 2015 testing dataset using Rossiello et al.'s summarizer and each of the baseline and test Word2Vec models, resulting in 30 summarizations produced by each of the six models. These summarizations were then compared to the human-made gold standard summaries using the ROUGE metric, version 2.1.2. Due to the significant impact the ROUGE parameters could have on the scores (the

same dataset can have a score from 4% to 26%, depending on the ROUGE parameters), all ROUGE parameters were left at their defaults, and the NLTK German stop word list was used for stop word filtering. The inclusion of synonyms was omitted due to the absence of support for the German language.

3.1.3 Turkish Summarization Experiment

In the Turkish Summarization experiment, the same procedure from the Summarization Experiment was performed on the database dump from the Turkish Wikipedia. In this experiment, no additional models were trained on portions of the corpus; instead, the only summarizations compared were those created on the full unstemmed Turkish corpus and the full stemmed Turkish corpus. The primary issue anticipated in the experiment was the stemmer, as low-resource languages generally do not have quality stemming algorithms (if any). Whether stemming helps or hinders summarizations is highly dependent upon the quality of the stemmer itself. Because NLTK does not have support for any stemmers in Turkish, GitHub's user-top-rated stemmer, Osman Tunçelli's Turkish Stemmer (2015), was used. The same approach was used for choosing the stop word list, Ahmet Aksoy's Turkish stop word list (2016). This list was used in order to implement the same ROUGE parameters as in the Summarization Experiment.

3.1.4 Word2Vec Model Accuracy Experiment

After careful examination of the results from the Summarization Experiment, it was clear that an additional experiment was needed to test whether the stemmed Word2Vec models were more semantically accurate than those comprising the baseline. Though an in-depth experiment testing the accuracy of the embedding models is beyond the scope of this paper, a small-scale experiment was performed to provide insight for future research.

In this experiment, a list of ten randomly-generated words[2]—including *nachhallen* ("reverberate"), *irische*, ("Irish"), and *Befehl* ("command")—was created. For every word on the list, the 5 closest embeddings (words deemed most similar in meaning by the model) were obtained from the model. Next the number of "related words" were counted. These were comprised of *antonyms,*[3] *hypernyms,*[4] *hyponyms,*[5] *co-hyponyms,*[6] *troponyms,*[7] and words deemed "related to" by Thesaurus.com. This process was then repeated for each of the six Word2Vec models from the Summarization Experiment (with the list being stemmed for the stemmed models). The number of "related words" output by each model were then compared.

3.2 Results

3.2.1 Summarization Experiment

Table 3.2 shows the average recall scores (in percentages) over all 30 summaries for each model in the experiment. Table 3.3 shows the percent improvement in scores from the baseline to the test summaries, and whether the improvement was calculated as statistically significant in a one-tailed t-test.

[2] https://randomwordgenerator.com
[3] An antonym is the opposite of a word (i.e. "dark" is an antonym of "light")
[4] A hypernym is a broader category of a word (i.e. "animal" is a hypernym of "dog")
[5] A hyponym is a more specific category of a word (i.e. "dog" is a hyponym of "animal")
[6] Co-hyponyms are words that share the same hypernym (i.e. "dog" and "cat" are co-hyponyms because they are both hyponyms of "animal")
[7] Troponyms are more specific verbs (i.e. "sprint" is a troponym of "run")

Table 3.2: ROUGE Recall Scores (%) from Summarization Experiment

Model	Rouge-1	ROUGE-2
Full German	5.29	0.36
Full Stemmed German	5.45	0.39
10% German	5.06	0.30
10% Stemmed German	5.17	0.37
1% German	5.07	0.28
1% Stemmed German	5.41	0.46

Table 3.3: Improvement (%) of Stemmed Models and Statistical Significance

Model Comparison	ROUGE	% Improvement of Stemmed Model	Significant at $p < .05$
Full German vs. Full Stemmed German	ROUGE-1	3.02%	X
	ROUGE-2	8.3%	X
10% German vs. 10% Stemmed German	ROUGE-1	2.17%	X
	ROUGE-2	23%	X
1% German vs. 1% Stemmed German	ROUGE-1	6.71%	X
	ROUGE-2	64.3%	✓

These scores indicate that every summary made—both from the stemmed training data and the unstemmed—shared an average of only 5% of the same words with the gold standard summary. At first glance, it doesn't appear as if stemming the training data has any significant effect on the summaries created. Though the ROUGE-2 scores of the 1% German Corpus were improved by 64%, the score only went up from 0.28% to 0.46%. This means that only an average of 0.28% of bigrams from baseline summarizations were also in the gold standard summary, and that only 0.46% of bigrams from the test summarizations were in the gold standard summary. Because of the small size of the testing set, this could very well mean that there was only *one* more bigram match in the test summaries, which is hardly an improvement.

3.2.1.1 Unstemmed vs. Stemmed ROUGE Scores

The results of the Stemming Experiment suggest that stemming had little to no impact on the embedding models, and that the summaries produced were nearly identical. To verify this, the ROUGE-1 test was re-implemented to compare the summaries from the stemmed model directly to the summaries from the unstemmed model (this time including stop words in the calculations). The ROUGE-1 and ROUGE-2 F-Scores (the average between precision and recall, represented as a percentage between 0 and 100) are in Table 3.4. In this test, the two sets of summaries were compared to each other to evaluate their average similarity, instead of one summary being scored on its similarity to the gold standard. Therefore, the F-Scores are presented instead of the recall scores. If the summaries were practically identical, as the summarization experiment results suggest, the scores should be close to 100%.

Table 3.4: ROUGE Word and Bigram Similarity (%) Between Summarizations from Stemmed and Unstemmed Models

Corpus size	Word Similarity (ROUGE-1)	Bigram Similarity (ROUGE-2)
Full German vs. Full Stemmed German	22%	7%
10% German vs. 10% Stemmed German	22%	6%
1% German vs. 1% Stemmed German	22%	6%

These results indicate that, although the baseline and test summaries were both almost equally similar/different from the gold standard summaries, they are not similar to each other, with an average of only 22% of the total words shared between the two—and that number drops to 7% with the removal of stop words.

3.2.1.2 Unstemmed vs. Stemmed Sentence Similarity

Because both summaries were created using an extractive summarizer on the same article, their similarity can best be measured by the number of sentences that appear in both. PrePost SEO's online Plagiarism Comparison Tool[8] was used to calculate the percentage of sentences that appeared in both the baseline summaries and the test summaries. Because of the very low chances of 30 articles on completely different topics having topic sentences in common, all baseline summaries were combined into a single file and compared to a file containing all the test summaries. The results of this test are in Table 3.5.

Table 3.5: Sentence Similarity (%) Between Summaries from Stemmed and Unstemmed Models

Corpus size	Sentence Similarity
Full German vs. Full Stemmed German	25%
10% German vs. 10% Stemmed German	28%
1% German vs. 1% Stemmed German	16%

The change in percentage is not drastic, but it does paint a slightly different picture. This shows that the similarity of the summaries made on the different models decreases as the corpora become smaller.

3.2.1.3 Full Corpus vs. 1% Corpus Sentence Similarity

Thus far the results have confirmed that stemming the training data did significantly change the output summary, but in the scope of this experiment did not

[8] https://www.prepostseo.com/plagiarism-comparison-search

improve the scores. The last test in the Summary Experiment investigated the sentence similarity between the summaries from the full models and the 1% models. This test was implemented to show the effects that shrinking the training data had on summarization output. In Table 3.6, the same Plagiarism Comparison Tool was used to detect sentence similarity of the summaries made on the full corpus to the summaries made on 1% of the corpus.

Table 3.6: Sentence Similarity (%) Between Summaries from the Full Corpus vs. the 1% Corpus

Models	Sentence Similarity
Full Corpus vs. 1% Corpus	27%
Full Stemmed Corpus vs. 1% Stemmed Corpus	31%

These results show that the summaries made on the larger corpora varied significantly from those made on the smaller corpora, showing that stemming the training data did make a difference in the summarization output.

3.2.2 Turkish Summarization Experiment

The results of the Turkish Summarization Experiment showed similar patterns to its German counterpart. Summaries created from stemmed models had only slightly higher ROUGE scores than those created from the unstemmed models, but the summaries themselves had only a 30% sentence similarity between the test and baseline. Table 3.7 shows the vocabulary size of the stemmed and unstemmed Turkish models, the percent decrease in vocabulary after stemming, as well as the average ROUGE recall scores and sentence similarity percentage (using the same plagiarism detector as before):

Table 3.7: Turkish Summarization Experiment ROUGE Scores and Sentence Similarity

Model	Vocabulary Size	Vocab. Size % Decrease	ROUGE-1 Recall Score (%)	Sentence % Similarity
Turkish	346,583	40%	4.69	30%
Stemmed Turkish	208,161		4.70	

3.2.3 Word2Vec Model Accuracy Experiment

Two tables were created from the results of the Word2Vec Model Accuracy Experiment: the first (Appendix A) is comprised of the original German results of this experiment, and the second (Appendix B) shows the English translations for each word. Table 3.8 shows the number of "related to" words given by each model, as well as the totals for the stemmed and unstemmed models.

Table 3.8: Fraction of "Related Words" Given by Model

Unstemmed Model	Number of "Related Words"	Stemmed Model	Number of "Related Words"
Full German	31 / 50	Full Stemmed German	36 / 50
10% German	24 / 50	10% Stemmed German	35 / 50
1% German	12 / 50	1% Stemmed German	20 / 50
TOTALS	**67 / 150**		**91 / 150**

CHAPTER 4

DISCUSSION

4.1 Summarization Experiment

The ROUGE scores showed that the baseline summaries and the stemmed summaries were equally different from the gold standard, but also very different from each other. In an attempt to get a better understanding of what these summaries *did* share with the gold standard, a program was written to create two lists of words shared between the gold standard summaries and the generated summaries (one for the baseline summaries and one for the test summaries), excluding stop words. The resulting lists showed four distinct categories of words.

The first were not actually words, but numbers. Most of the human-made gold standard summaries listed one or two significant dates concerning the topic. The embedding model treats numbers (such as years) as words, and thus they receive their own TFIDF weight. Individual years are relatively low-frequency "words" in the language model, meaning they are given relatively high TFIDF scores when appearing in articles, and are thus more likely to be chosen for the summary.

The second category of words were those describing the main topic. As would be expected in a summary, most of the generated summaries had at least one sentence naming the topic of the article. Thus, many of the words shared between the summaries were names, such as "Pink" and "Floyd," "Ludwig," "Saxony," and "Dresden." The third category of words shared between the generated summaries and the gold standard were the

insignificant words. These words, while not included in NLTK's German stop word list, were not significant topic words and did not alone relay crucial information about the article. These words included many prepositions, conjunctions, and other common words that did not generally contribute essential content to the summary, such as *nach* ("after"), *neu* ("new"), and *kurz* ("short").

The last category of words were the significant descriptor words. Had the generated summaries been much more like the gold standard summaries, this category would have been the biggest; but because the summaries differed greatly, words of this category—including *Singvogel* ("songbirds"), *Gehirn* ("brain"), and *Schildkröte* ("turtle")—were the least encountered on the list. These results gave no indication of which summaries were superior, as both lists were equally distributed with the four categories of words mentioned above.

4.2 Word2Vec Model Accuracy Experiment

The most significant question remaining was, "how did embedding models given the exact same raw training corpus produce such different extractive summaries?" The stemming of one corpus resulted in a significantly smaller (17% - 18%) vocabulary count; and in theory, the embeddings for the stems had to split the difference between what would have been the embedding spaces of all the words they now encompass. A simplified illustrative example of this is the word *fliegend*. In German, this means "flying," and in the full unstemmed embedding model, its embedding is mapped near "hovering," "leaping," and "fluttering." However, in the stemmed model, the word *fliegend* (which is itself a stem) encompasses every word in the raw corpus that has the same stem. This includes *Fliegender*, which is someone who sells vegetables. The words have nothing to do with

each other semantically but happen to share the same stem within the Snowball stemmer. Therefore, the embedding in the stemmed model for *fliegend* is in a very different part of the vector space than its counterpart in the unstemmed model, because it must split the difference between all the meanings it now represents, including "flying" and "vegetable seller." The results are two embedding models that differ significantly in "shape." Therefore, the placement of the centroid of a document and the embeddings for the sentences of that document differ between the models as well, thereby creating very different extractive summaries with the centroid-based summarization method.

4.2.1 Difference Between Embedding Models

With reasonable proof that neither the baseline nor test summaries were any closer to the gold standard summary, this thesis was left without an objective answer as to which summary was better in the scope of this experiment. However, there was still the question of what qualitative effect stemming had on the Word2Vec embedding models, and if either the baseline or test models were objectively more accurate. The Word2Vec Model Accuracy Experiment was performed to test this.

Besides generally producing more accurate embeddings, the stemmed models also had the distinct advantage of a more inclusive vocabulary after stemming. For example, one of the words fed into the embeddings was *nachhallen*, which means "reverberate." The 10% and 1% unstemmed models did not have this word in their vocabulary and could therefore give no output. The equivalent stemmed models, however, did have *nachall*, the stem of *nachallen*, which encompassed *nachhallend* (resonating, resonant, or reverberative), *Nachhallkurve* (reverberation curve), and *Nachhallraum* (echo chamber). Though the stemmed models were trained with the same corpora (meaning they were not

trained with the word *nachhallen* either), the stemming of the corpus allowed the model to deduce the meaning of the word by its shared stem.

In this small-scale qualitative analysis, the models trained on stemmed datasets were shown to be the more accurate and more inclusive models, though more in-depth and encompassing research is needed.

4.3 Summarization Experiment Limitations

Though the Summarization Experiment was a relatively straightforward one, there were many options in terms of the methodology. The aim of the methodology choices in this paper was to keep the experiment as simple as possible so the results could be informative. While most of the methodology was laid out beforehand with Rossiello et al.'s paper—including the language, summarizer, dataset, and metric—the methodology choices which differed from the original work were chosen to either test the effects of stemming on automated summarization or minimize the overall complexity of the experiment.

Though an in-depth analysis of the summarization method used in Rossiello et al.'s paper was planned for this thesis, the experiment limitations and their effects on the results made it impossible to draw any conclusions about the quality of the summarizer itself.

4.3.1 The MultiLing 2015 Dataset

4.3.1.1 Dataset Size

Possibly the most limiting factor in this experiment was the dataset used. The MultiLing 2015 dataset is a popular choice for testing summarizers and is often one of the only summarization corpora in a language. While interest in automatic text summarization

engines is growing rapidly, the number of datasets to test them on remains extremely low for languages other than English.

The dataset contains only 30 Wikipedia articles for each language, and only one gold standard summary for each article. This extremely small size of testing data leads to significant limitations on the insight it can provide. The summarizer (if it is embedding driven) must have adequate mappings of the 30 specific topics the articles covered (topics range from a species of tree to Max Lieberman to Pink Floyd). If the embedding model performs poorly on just one of the summaries, it can significantly impact the average scores.

4.3.1.2 Gold Standard Summary

The MultiLing dataset contains only one gold standard summary for each article. Because there is no objective "best" summary for any given text, it is difficult to compare a summary to a single gold standard without being subjective. Extractive summarization occupies a specific role in summarization and is not expected to perform outside of its capabilities. Extractive summarization is created by choosing the best sentences (or sometimes phrases) from a document to create as informative a summarization as possible in a limited number of sentences, words, or characters. These summarizers are both easier to write and quicker to implement; and in many cases perform objectively well enough for the task at hand. However, extractive summarization does not and cannot compete with a traditional summary made by a human, as humans do not generally use the extractive process when creating summaries.

Abstractive summarization (the other summarization school of thought mentioned in the literature review) has the goal of creating summaries that are as close to human-made

summaries as possible and are intended to use the same fundamental thought process as humans. These summarizers are more theory than reality at the moment, because mimicking the human thought process is an extremely challenging task and has been the foremost goal of many computer sciences (such as neuro-linguistic programming) since their inception.

However, extractive summarization should not be seen as simply worse than abstractive. Instead, it should be understood that they serve different purposes. Extractive summarization's purpose is to provide the main idea of the article in a computationally cheap and effective manner. Therefore, comparing extractive summarizations to traditional human-made ones is both uninformative and ineffective. Instead, extractive summarizations should be compared to ideal extractive summarizations. Currently, the simplest method of creating these ideal extractive summarizations would be tasking (multiple) humans with picking the *n* best sentences from an article, thereby creating a human-made gold-standard extractive summarization for comparison.

4.3.2 The ROUGE Metric

In this experiment, it is hard to draw the line between the limitations of the testing dataset and the limitations of the testing metric. Because the gold standard summary was not made with sentences from the original article, the automatically generated summary is already at a disadvantage in the ROUGE metric. In addition, synonyms are not given credit by ROUGE (unless the language is one of the three supported), stacking the odds even higher against the summarization.

4.3.2.1 ROUGE Parameters

ROUGE is a very superficial metric, which is certainly its most common criticism from researchers. However, another substantial criticism of ROUGE's effectiveness is its large number of parameters, which can have a great impact on the final scores. In addition to the multitude of different ROUGE scoring types (i.e. ROUGE-1, ROUGE-2, ROUGE-SU, etc.), one can change whether stop words are counted in the precision and recall percentages; if so, the user must provide their own list of stop words, which can be any length or combination of words. If the language being used is supported by one of ROUGE's available stemmers, one can choose whether to count words with the same stem as matches. If there is POS tagging available in the language being used, one can also choose whether to count synonyms as matches. The POS taggers rely on Stanford's NLP system, but the only supported languages as of April 2018 are English, Chinese, and Arabic. One can even change the balance of the F-score to favor or even copy either the precision or the recall. Previous versions of ROUGE had even more parameters that could be tweaked to alter the final scores.

Any of these parameters can have a huge impact on the scores. In the Summarization Experiment, the same summaries could obtain ROUGE scores as high as 26%, or as low as 4%, depending entirely on the stemming and stop words parameters. The scores could likely be varied even more by introducing different stop words lists, or using the POS tagging. While it is generally expected that researchers provide their ROUGE parameters in their papers, it is most often listed as an afterthought in a footnote instead of treated as a significant part of the experiment process. Researchers also do not generally state which ROUGE score they are using (between recall, precision, or F-score), which is

an important fact to relay. The most popular choice is the recall score alone, but the F-Score is also used in experiments where the length of the summary is not specifically limited.

4.3.2.2 Suggestions for Improvement

While ROUGE was certainly an impressive and effective metric compared to its peers at the time, it has been over 14 years since its release; and it can be argued that a more effective metric is sorely needed. Newer ROUGE adaptations—such as ROUGE-WE—show significant promise, but must be developed to accept languages other than English to be considered contenders on an international level.

To assess the similarity of two extractive summaries, as in Tables 3.5 and 3.6, the plagiarism detection tool proved to be much more effective in determining the percentage of sentences that were the same. Sentence scoring and ranking is an idea as old as automatic summarization itself, but it has yet to be effectively applied to summarization *evaluation*, except in Radev and Tam's manual summary evaluation system (2003). Granted, because the sentence scoring method has yet to produce perfect extractive summaries, it cannot be expected to independently evaluate and score summaries. Until these methods advance significantly, it is important for researchers to treat each sentence of an extractive summary as a single part that makes up a whole (the article), and not to perform word-level topic analysis as would be expected for an abstractive summary.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

In the age of big data, quantity is king. Researchers in NLP often strive to increase the quantity of their linguistic data; and languages with low quantities of data are significantly less researched. NLP—especially word vectorization—is greedy in its insatiable need for data. However, a large quantity of linguistic data is not always available; and for applications in many languages, linguistic data is scarce. In the absence of quantity, quality must be made a priority. As the experiments in this paper have shown, improving the quality of what little linguistic data is available can have a considerable impact on NLP output. These experiments illustrated how decreasing the quantity of Word2Vec vocabulary and increasing the quality of the embeddings led to improved semantic accuracy of the model.

The similarity of the Turkish results to those of the German Summarization Experiment suggest that the Turkish stemmed embedding model experienced the same semantic improvements. This also suggests that stemming could improve embedding models for other low-resource, morphologically complex languages. More accurate embedding models can help improve the results of NLP applications in low-resource languages, which in turn can help businesses operating in these languages compete in the global marketplace. Improved sentiment analysis can provide these companies a better understanding of their customer base; and more accurate product recommender engines can considerably increase sales. Product labeling and international communication can also

benefit greatly from improved translation engines. More accurate word embedding models have the potential to improve NLP tasks in every corner of the global marketplace.

There are many opportunities for future research in the areas described in this paper. Primarily, more extensive semantic testing is needed on Word2Vec models from stemmed and unstemmed corpora. Future research in improving embedding models through stemming should expand to other languages and applications. Research in improving stemming and lemmatization algorithms is also sorely needed for low-resource, morphologically complex languages. As well, NLP is in need of a modern and accurate evaluation metric created specifically for extractive summarizations, in which summaries are evaluated on the quality of the sentences chosen, and not compared to traditional human-made summaries. Lastly, further research is needed in improving both the quantity and the quality of linguistic data in resource-poor languages.

APPENDIX A

MOST-SIMILAR WORDS LIST FOR STEMMED AND UNSTEMMED GERMAN EMBEDDING

Word	Full German	Full STEMMED German
Hund	katze	Aff
	dackel	dackel
	haustier	esel
	affe	kaninch
	herrchen	elefant
religionen	religion	weltanschau
	glaubensvorstellungen	ethik
	glaubensrichtungen	glaubensinhalt
	kulturen	glaubensvorstell
	glaubensformen	spiritualitat
fliegend	rennend	tollkuhn
	beutetier	mow
	rüttelflug	dressiert
	hüpfend	delphin
	flatternd	jagend
Irisch	gälisch	walis
	walisisch	schottisch
	dún	nordir
	schottischgälisch	englisch
	baile	britisch
nachhallen	lebest	klang
	himmel	horeindruck
	feroce	klangeindruck
	kreuzritterschwert	klangcharakt
	klugem	zusammenklang
beginnen	vorbereiten	end
	angefangen	anfang
	entschließen	gunst
	vollziehen	beend
	beenden	ausbruch
Vordenker	wegbereiter	verfecht
	verfechter	wegbereit
	theoretiker	hauptvertret
	vorkämpfer	grundervat
	hauptvertreter	wortfuhr
kooperieren	zusammenarbeiten	kommunizi
	kooperierten	kooperiert
	zusammenzuarbeiten	zusammenzuarbeit
	agieren	kooperier
	beteiligen	konkurri
Befehl	anweisung	anweis
	weisung	einsatzbefehl
	befehle	weisung
	angriffsbefehl	ruckzugsbefehl
	kommando	oberbefehl
Kultur	volkskultur	kulturell
	alltagskultur	kulturgeschicht
	kulturelle	volkskultur
	kunst	religion
	kulturgeschichte	alltagskultur

	10% German	10% Stemmed German
Hund	katze	aff
	vampir	pferd
	fährte	kafig
	kater	kaninch
	käfig	katz
religionen	religion	religios
	traditionen	mystik
	hinduismus	weltanschau
	riten	spiritualitat
	völker	christlich
fliegend	bruthöhle	aufblasbar
	schwimmend	schwimmend
	tracheen	flieg
	tauchend	gepanzert
	mycel	flugkorp
irische	anglisiert	walis
	gälisch	schottisch
	cill	britisch
	schottischgälisch	austral
	walisisch	englisch
nachhallen	X	klang
	X	ton
	X	witz
	X	wasserstrahl
	X	auftriebsgewinn
beginnen	beginnt	end
	begonnen	anfang
	begannen	ausbruch
	konzentrieren	mitt
	gehen	gunst
Vordenker	wegbereiter	wegbereit
	theoretiker	hauptvertret
	denker	verfecht
	hauptvertreter	grundervat
	begründer	begrund
kooperieren	vernetzt	kommunizi
	kooperierten	kooperiert
	interessengruppen	koordini
	fusionieren	zusammenzuarbeit
	koordinieren	interagi
Befehl	kommando	anweis
	anweisung	oberbefehl
	oberbefehl	kommando
	angriff	angriff
	kapitulation	weisung
Kultur	wissenschaft	kulturgeschicht
	kunst	kulturell
	kulturgeschichte	religion
	alltagskultur	volkskultur
	kulturen	alltagskultur

	1% German	1% Stemmed German
hund	gerne	madch
	ruft	schmerz
	fragt	katz
	tanzen	leut
	gefühle	korp
religionen	religion	christlich
	buddhismus	religios
	begriffe	rhetor
	verständnis	kultur
	betrachtung	tradition
fliegend	wasseroberfläche	winzermess
	vulkanischen	3msynchronspring
	wellig	weich
	warme	muschelform
	wassers	gewolbt
Irische	dun	britisch
	kunstwort	austral
	italischen	walis
	zitrusfrüchte	griechisch
	slowenisch	usamerikan
nachhallen	X	hormon
	X	amin
	X	l2
	X	ammoniak
	X	zwischenprodukt
beginnen	fällt	end
	wechseln	anfang
	ziehen	ausbruch
	erwacht	mitt
	fallen	während
Vordenker	ramasami	mussolinis
	dravidischen	unitari
	willensbildung	tsūshinsha
	ehrenamt	lascell
	parlamentsabgeordneten	sportbeweg
kooperieren	initiativen	gemeinschaftsforsch
	privatpersonen	lukrativ
	kooperiert	konkurrier
	externe	medizintechn
	forschungsprojekte	kooperiert
Befehl	heer	oberbefehl
	gefecht	grenadi
	kommando	anweis
	mecklenburger	attentat
	offizieren	generalmajor
Kultur	erforschung	gesellschaft
	kunst	kulturgeschicht
	kulturellen	kunst
	forschung	kulturell
	themen	religion

APPENDIX B

MOST-SIMILAR WORDS LIST FOR STEMMED AND UNSTEMMED GERMAN EMBEDDING: ENGLISH TRANSLATIONS

	Full German	Full STEMMED German
dog	cat	(stem of) monkey
	daschund	daschund
	pet	donkey
	monkey	rabbit
	master	elephant
religions	religion	ethics
	notions	(stem of) ideological or worldview
	directions	(stem of) beliefs
	cultures	(stem of) innermost beliefs
	faith	spirituality
flying	racing	foolish or daredevil
	prey (animal)	(stem of) seagull
	hovering	trained
	leaping	dolphin
	fluttering	hunting
Irish	Gaelic	(stem of) Welsh
	Welsh	Scottish
	(Irish) close*	(stem of) Northern Ireland
	Scotts Gaelic	English
	(Irish) town*	British
reverberate	may live	sound
	sky	auditory sensation
	(Italian) fierce	sounded impressive
	crusader's sword	character of sound
	clever	chord
begin	to prepare	end
	started	beginning
	decide	favor
	make	end all
	break up	outbreak
mastermind	forerunner	(stem of) advocate or defensible
	advocate	(stem of) pioneer
	theorist	(stem of) chief agent
	champion	(stem of) founder
	all representatives	(stem of) spokesperson
cooperate	work together	(stem of) to communicate
	cooperated	cooperates
	together	working together
	act	(stem of) to concur
	participate	(stem of) to compete
command	instructions	(stem of) to instruct
	instruction	operational command
	commands	instruction
	attack command	retreat command
	command	supreme command
culture	popular culture	cultural
	everyday culture	cultural historian
	cultural	popular culture
	art	religion
	culture history	everyday culture

	10% German	10% Stemmed German
dog	cat	(stem of) monkey
	vampire	horse
	track	cage
	hangover	rabbit
	cage	cat
religions	religion	religious
	traditions	mysticism
	Hinduism	(stem of) ideological or worldview
	rites	spirituality
	international	Christian
flying	breeding burrow	inflatable
	floating	floating
	tracheas	fly
	diving	armored
	mycelium	missile
Irish	anglicized	(stem of) Welsh
	Gaelic	Scottish
	(Irish) cell*	British
	Scotts Gaelic	English
	Welsh	austral (having to do with the south)
reverberates	X	sound
	X	(stem of) volume or tone
	X	joke
	X	waterjet
	X	boost profit
begin	starts	end
	began	beginning
	started	outbreak
	focus	(stem of) midday
	go	favor
mastermind	forerunner	(stem of) pioneer
	theorist	chief agent
	thinker	(stem of) advocate or defensible
	all representatives	founder
	founder	(stem of) founded or justified
cooperate	networked	(stem of) to communicate
	cooperated	cooperates
	interest groups	(stem of) to coordinate
	merge	work together
	coordinate	(stem of) to interact
command	command	(stem of) to instruct
	instructions	supreme command
	supreme command	command
	attack	attack
	surrender	instruction
culture	science	cultural historian
	art	cultural
	culture history	religion
	everyday culture	popular culture
	cultures	everyday culture

	1% German	1% Stemmed German
dog	like	(stem of) girl / girls
	calls	pain
	asks	cat
	dance	(stem) of people
	feelings	(stem of) body
religions	religion	Christian
	Buddhism	religious
	concepts	(stem of) rhetoric
	understanding	culture
	viewing	tradition
flying	water surface	(stem of) small knife
	volcanic	(stem of) 3m synchronized jumping
	wavy	soft
	warm	shell-shaped
	water	domed
Irish	(Irish) close*	British
	coinage or artificial word	austral (having to do with the south)
	Italian	(stem of) Welsh
	citruses	Greek
	Slovenian	(stem of) U.S. American
reverberate	X	hormone
	X	(stem of) amino acids or anime
	X	12
	X	ammonia
	X	(stem of) intermediates
Begin	falls	end
	switch	beginning
	pull	outbreak
	awakes	(stem of) midday
	fall	while or meanwhile
mastermind	Ramasami (former Indian secretary of technology)	Mussolini
	Dravidian (family of languages in South Asia)	(stem of) unitarian
	willed education	(Japanese for) a news agency
	volunteering	sports movement
	parliament deputies	(stem of) to compete
cooperate	initiatives	community research
	private persons	lucrative
	cooperates	(stem of) to compete
	external	(stem of) medical engineering
	research projects	cooperates
command	army	supreme command
	battle	(stem of) soldier
	command	(stem of) to instruct
	Mecklenburger (breed of horse)	attack
	officers	major-general
Culture	exploration	society
	art	cultural historian
	cultural	art
	research	cultural
	subjects	religion

Unstemmed Models	Stemmed Models
67	91

*While not on the related words list, they counted to give the unstemmed model the benefit if the doubt

REFERENCES

Aksoy, A. (2016, October 7). *trstop*. Retrieved from GitHub: https://github.com/ahmetax/trstop

Allahyari, M., Pouriyeh, S., Assefi, M., Safaei, S., Trippe, E. D., Guitierrez, J. B., & Kochut, K. (2017). Text Summarization Techniques: A Brief Survey. *arXiv*.

Attardi, G. (2017, March 8). *WikiExtractor.py*. Retrieved from https://github.com/attardi/wikiextractor

Baumann, P., & Pierrehumbert, J. (2014). Using Resource-Rich Languages to Improve Morphological Analysis of Under-Resourced Languages. *In Proceedings of 9th International Conference on Language Resources and Evaluation (LREC'14)*.

Bolukbasi, T., Chang, K.-W., Zou, J., Saligrama, V., & Kalai, A. (2016). Man is to Computer Programmer as Woman is to Homemaker? Debasing Word Embeddings. *arXiv*.

Boukkouri, H. E. (2017, July 31). *A non-NLP application of Word2Vec*. Retrieved from Towards Data Science: https://towardsdatascience.com/a-non-nlp-application-of-word2vec-c637e35d3668

Bureau of Educational and Cultural Affairs Exchange Programs. (n.d.). *Critical Language Scholarship Program*. Retrieved from United States Department of State: https://exchanges.state.gov/us/cls

CNN. (2011, September 30). *U.S. military personnel by country*. Retrieved from CNN News: http://www.cnn.com/interactive/2012/04/us/table.military.troops/

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., & Harshman, R. A. (1990). Indexing by Latent Semantic Analysis. *JASIS 41*, 391-407.

Department of Linguistics at Ohio State University. (2011). Morphology. In *Language Files* (pp. 148-194). Columbus, OH: The Ohio State University Press.

Dholakia, R. (2014). *International relations rely on language communication.* International relations rely on language communication. .

Dunning, T. (1993). Accurate Methods for Statistics of Surprise and Coincidence. *Computational Linguistics*, 61-74.

Duong, L. (2017). *Natural language processing for resource-poor languages.* Retrieved from University of Melbourne Library: https://minerva-access.unimelb.edu.au/handle/11343/192938

Firth, J. (1957). A synopsis of Linguistic Theory 1930-1955. In *Studies in Linguistic Analysis* (pp. 1-32). Oxford: Oxford Philogical Society.

Franklin, L. (2013, October 6). Americans Suffer From Inadequate Foreign Language Education. *The Daily Texan*.

FRESA 2.1 FRamework for Evaluationg Summaries Automatically. (n.d.). Retrieved from Traitement utomatique de la Langue Naturelle Ecrite: http://fresa.talne.eu/

Gambhir, M., & Gupta, V. (2017). Recent automatic text summarization techniques: a survey . *Artificial Intelligence Review*, 1-66.

Hamilton, W. L., Clark, K., Leskovec, J., & Jurafsky, D. (2016). Inducing Domain-Specific Sentiment Lexicons from Unlabeled Corpora. *arXiv*.

Harabagiu, S., & Lacatusu, F. (2005). Topic Themes for Multi-Document Summarization. *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 202-209.

Haryalesmana, D. (2016, January 28). *ID-Stopwords*. Retrieved from GitHub: https://github.com/masdevid/ID-Stopwords

He, T., Chen, J., Ma, L., Gui, Z., Li, F., Shao, W., & Wang, Q. (2008). ROUGE-C: A Fully Automated Evaluation Method for Multi-document Summarization. *IEEE International Conference on Granular Computing*, 269-274.

International Telecommunications Union. (2017). *ICT Facts and Figures 2017* . Retrieved from Itu.int: https://www.itu.int/en/ITU-D/Statistics/Pages/facts/default.aspx

Jaech, A., & Ostendorf, M. (2015). Leveraging Twitter for Low-Resource Conversational Speech Language Modeling. *arXiv*.

Kheyrollahi, A. (2015). *Five Crazy Abstractions My Deep Learning Word2Vec Model Did*. Retrieved from Byte Rot: http://byterot.blogspot.in/2015/06/five-crazy-abstractions-my-deep-learning-word2doc-model-just-did-NLP-gensim.html

Kholy, A. E. (2016). *Pivot-based Statistical Machine Translation for Morphologically Rich Languages.* Columbia University Press.

Koehn, P., & Knight, K. (2003). Empirical methods for compound splitting. *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 1.* .

Kumar, E., & Samuels, E. (2015). *Spotify's music Recommendations Lambda Architecture*. Retrieved from slideshare.net: https://www.slideshare.net/eshvk/spotifys-music-recommendations-lambda-architecture

Lin, C.-Y. (2004). ROUGE: A Package for Automatic Evaluation of Summaries. *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, 74-81.

Lloret, E., Plaza, L., & Aker, A. (2017). The Challenging Task of Summary Evaluation: An overview. *Language Resources & Evaluation*.

Lovins, J. (1968). Development of a Stemming Algorithm. *Mechanical Translation and Computational Linguistics*.

Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. *IBM Journal*, 159-165.

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval.* Cambridge University Press. Retrieved from https://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013b). Efficient Estimation of Word Representations in Vecor Space. *arXiv*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Den, J. (2013a). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems,*, 3111-3119.

Moody, C. (2015). *A Word is Worth a Thousand Vectors*. Retrieved from Stitchfix.com: https://multithreaded.stitchfix.com/blog/2015/03/11/word-is-worth-a-thousand-vectors/

Nakov, P., & Ng, H. T. (2011). Translating from Morphologically Complex Languages: A Paraphrase-Based Approach. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, 1298-1307.

Nakov, P., & Ng, H. T. (2012). Improving Statistical Machine Translation for a Resource-Poor Language Using Related Resource-Rich Languages. *Journal of Artificial Intelligence Research*, 179-222.

Nenkova, A., & Vanderwende, L. (2005). The Impact of Frequency on Summarization. *Microsoft Research*.

Ng, J.-P., & Abrecht, V. (2015). Better summarization evaluation with word embeddings for rouge. *: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1925-1930.

Ng, J.-P., & Abrecht, V. (2015, September 24). *ROUGE-WE*. Retrieved from Github: https://github.com/ng-j-p/rouge-we

Oracle. (n.d.). *Boost Your Database Performance 10x with Oracle SecureFiles*. Retrieved from Oracle.com: http://www.oracle.com/technetwork/database/performance/boost-your-database-performance-10x-130376.pdf

Owczarzak, K. (2009). Dependency-based evaluation for automatic summaries. . *Proceedings of the joint conference of the 47th annual meeting of the ACL and the 4th international joint conference on natural language processing of the AFNLP*, 190-198.

Packham, S. (2016). *Crowdsourcing a Text Corpus for a Low-Resource Language.* Retrieved from OpenUCT: https://open.uct.ac.za/handle/11427/20436

Plottingman. (2015, June 10). *Top Languages of the Internet, Today and Tomorrow*. Retrieved from Unbabel: https://unbabel.com/blog/top-languages-of-the-internet/

Radev, D. R., & Tam, D. (2003). Single-document and multi-document summary evaluation via relative utility. *Proceedings of the twelfth international conference on Information and knowledge management*.

Radev, D. R., Jing, H., & Budzikowska, M. (2004). Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. *Information Processing and Management*, 919-938.

Robbani, H. A. (2017, October 28). *PySastrawi*. Retrieved from GitHub: https://github.com/har07/PySastrawi

Rossiello, G., Basile, P., & Sameraro, G. (2017). Centroid-based Text Summarization through Compositionality of Word Embeddings. *Proceedings of the MultiLing 2017 Workshop on Summarization and Summary Evaluation Across Source Types and Genres*, 12-21.

Rossiello, G., Basile, P., & Semeraro, G. (2017, March 7). *text-summarizer*. Retrieved from Github: https://github.com/gaetangate/text-summarizer

Ruder, S. (2018). *Word Embeddings in 2017: Trends and Future Directions*. Retrieved from Ruder: http://ruder.io/word-embeddings-2017/

Sjöbergh, J. (2007). Older versions of the ROUGEeval summarization evaluation system were easier to fool . *Information Processing and Management: an International Journal*, 1500-1505.

TAC 2008 Opinion Summarization Task Guidelines. (2008). Retrieved from National Institute of Standards and Technology: https://tac.nist.gov//2008/summarization/op.summ.08.guidelines.html

The Central Intelligence Agency. (2017, Feb 09). *Foreign Language*. Retrieved from The Central Intelligence Agency: https://www.cia.gov/careers/foreign-language

The Defense Intelligence Agency. (n.d.). *Careers*. Retrieved from Defense Intelligence Agency: http://www.dia.mil/Careers/Foreign-Languages/

Tratz, S., & Hovy, E. (2008). Summarization Evaluation Using Transformed Basic Elements. *Proceedings of the 1st Text Analysis Conference*.

Tripodi, R. (2018). *Visualizing Italian Word Embeddings*. Retrieved from Rocco Tripodi: www.roccotripodi.com/wp-content/uploads/2017/11/projector.png.

Tsarfaty, R., Seddah, D., Goldberg, Y., Kübler, S., Candito, M., Foster, J., . . . Tounsi, L. (2010). Statistical Parsing of Morphologically Rich Languages (SPMRL) What, How and Whither. *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages,*, 1-12.

Tunçelli, O. (2015, January 11). *turkish-stemmer-python*. Retrieved from Github: https://github.com/otuncelli/turkish-stemmer-python

Vector Representations of Words. (2018, March 29). Retrieved from TensorFlow: https://www.tensorflow.org/tutorials/word2vec

Weißweiler, L., & Fraser, A. (2017). Developing a Stemmer for German Based on a Comparative Analysis of Publicly Available Stemmers. *GSCL*.

Wong, K.-F., Wu, M., & Li, W. (2008). Extractive Summarization Using Supervised and Semi-supervised Learning. *Proceedings of ACL 2008*, 985-992.

Zou, W. Y., Socher, R., Cer, D., & Manning, C. D. (2013). Bilingual Word Embeddings for Phrase-Based Machine Translation. *ai.stanford.edu*.

BIOGRAPHICAL INFORMATION

Kalen Goss Manshack graduated with an Honors Bachelor of Arts in Linguistics in May 2018 from the University of Texas at Arlington. She is pursuing a career in Computational Linguistics and Natural Language Processing, and plans to obtain her master's degree in Computational Linguistics from the University of Washington. In additional to research in automatic text summarization for resource-poor languages, she has also performed research in sentence compression.