Final Year Project Interim Report

A Word Lexicon Builder Using Neural Word Embeddings

Daire Murphy - 15441458

Supervisor: Derek Greene

A thesis submitted in part fulfilment of the degree of
BSc. (Hons.) in Computer Science with Data Science

UCD School of Computer Science
University College Dublin

Table of Contents

List of Figures

List of Important Abbreviations Used Within

TLA	*Three Letter Acronym*
FLW	*Four Letter Word*
PFO	*Public Funding Option*

Abstract

Word lexicons are commonly used to systematically filter and search large text corpora, to isolate all documents that are related to the selected concept of interest. Using a combination of pre trained word embeddings, as well as self-developed word recommendation methods, this project evaluates and compares the word recommendations produced from each of these models. This data is examined to identify the most accurate models to be used and an interactive web interface is implemented, using the purposed word embeddings to create a lexicon recommendation method. This web interface allows for editing of recommended words, improving accuracy of the models for users. Visualizations of the embedding spaces are added to the interface to allow exploration by users of the word embedding models used within this project.

Project Specification

Word embeddings, which refer to a set of language modeling and feature learning techniques in natural language processing algorithms that map words or phrases from a vocabulary to vectors of real numbers, have over the last number of years been used in the creation of word lexicons, a list of words interested to a specific theme or concept, on large text corpora. Word embedding models are trained by applying a neural network to a large text dataset, and such many pretrained models are available on an extreme variety of datasets.

A primary focus of this project was the creation, development and evaluation of several word recommendation methods, along with further comparison and evaluation of these models with other pretrained models. The implementation of these models to an interactive web interface is required to create an online lexicon recommendation method. The student will then be expected to add interactive visualizations to the web interface to allow exploration of the embedding spaces by users

Core:

- Develop a word recommendation method, based on a given seed set and a single word embedding model.
- Perform an evaluation to compare the word recommendations produced using embedding models trained on different text datasets and using different algorithms.
- Implement an interactive web interface for lexicon building, which uses the proposed lexicon recommendation method. This interface should allow users to accept or reject recommended words, in order to produce a more useful final lexicon.
- Develop an “ensemble” word recommendation method, which combines the outputs from different embeddings into a single set of recommendations. Incorporate this new method into the web interface.

Advanced:

- Add interactive visualisation functionality to the web interface to allow users to explore the embedding spaces in more detail.
- Design and conduct a user study to assess the usefulness of the web interface and the lexicons which it produces.
- Perform an evaluation to examine the extent to which embedding algorithms can produce different results when applied to the same data.

Chapter 1: **Introduction**

Traditionally, the creation of word lexicons was a hugely intensive manual process, requiring a human to select a small set of highly relevant seed words, which are then perceived as central to the concept. Through the process of trial and error, lexicons could be expanded until the lexicon was satisfactory. This was undoubtedly a subpar method of text analysis, due to the time-consuming nature of this process and the ad-hoc selection of seed words combining to make any work difficult to justify and challenging for other researchers to replicate.

Recently, to address these issues, work has been done looking at the use of word embeddings, which are a learned representation for text where words that have the same meaning have a similar representation, for recommending relevant words when building a lexicon [1]. Word embeddings are in fact a class of techniques where words are represented in a predefined vector space as real-valued vectors. Each word is mapped to one vector in this vector space, often tens or hundreds of dimensions. This means words which frequently appear in similar context in the original text will appear close together in the vector space, while words that do not frequent together will be dissimilar.

In this project we will look to train an embedding model using a variety of large text datasets and popular algorithms such as Word2Vec [2] and FastText [3]. Using these, we will implement our embedding model on a new web-based tool for constructing word lexicons in an interactive and efficient manner. As the processes of using word embeddings can produce diverse results, different models will be evaluated and compared, and a method for combining the results of different models into a single set of recommendations will be used when creating the lexicon.

1.1 Motivations

My motivations for this project derived from reading papers that employed the use of word embeddings to create lexicons, such as Curatr: A Platform for Semantic Analysis and Curation of Historical Literary Texts [1]. In this work, a platform ‘Curatr’ which “uses neural word embeddings to build conceptual lexicons specific to a given theme or topic” was created for the exploration and curation of literature with machine learning-supported semantic search. This platform was developed based on 35918 English language digital texts from the British Library and uses the *Word2Vec* approach.

This was interesting to me as this platform was using a huge text corpus of old texts found in the British library for its word embeddings and led me to research into how different data sets, as well as different embedding models might produce largely different lexicons. This further interested me in finding a suitable method of combining ranked lists of word recommendations provided by different models and datasets to produce an improved lexicon.

1.2 Project Objectives

In this section I outline the direction of my project with a list of objectives, highlighting what I hope to accomplish from the exploration and analysis of this topic and implementation of my work.

- Obtain a number of appropriate large text datasets of varying size and content.
- Create several different word embedding models from these datasets using a combination of popular algorithms.
- Compare and evaluate these models with already available pre-trained models.
- Develop an ensemble recommendation method, combining the outputs of several methods into a single set of recommendations.
- Implement an interactive web interface for lexicon building, using the proposed lexicon recommendations.

1.3 Summary of Report

This report has been split into three sections: related work, progress to date and future work. The first section, related work, discusses three aspects of the project and looks at works done that closely relate to these aspects. This is followed by my current progress to date, which details all the progress I have the project so far, including some visualizations of the embedding models. Lastly, this report will cover any future work intended for this project and propose a plan in which this work will be completed.

Chapter 2: **Related Work**

2.1 Text Analysis

2.2 Word Embeddings

The term word embeddings were originally coined by Bengio et al. in 2003 [3], who referred to word embeddings as "distributed representations of words", and used a neural language model along with the model's parameters to train these embeddings. However, it was arguably Collobert and Weston who were the first to demonstrate the power of pre-trained word embeddings in their paper "A unified architecture for natural language processing" [4], back in 2008. This paper established word embeddings as useful tool for downstream task as well as introduces neural network architecture that forms the basis for several future works, and foundation for many current approaches. Initial word embedding models consisted of feed-forward neural networks that would take words from a vocabulary as input, embeds them as vectors into a lower dimensional space, uses through back-propagation as fine tuning, yielding word embeddings as weights of the first layer, known as the Embedding layer.

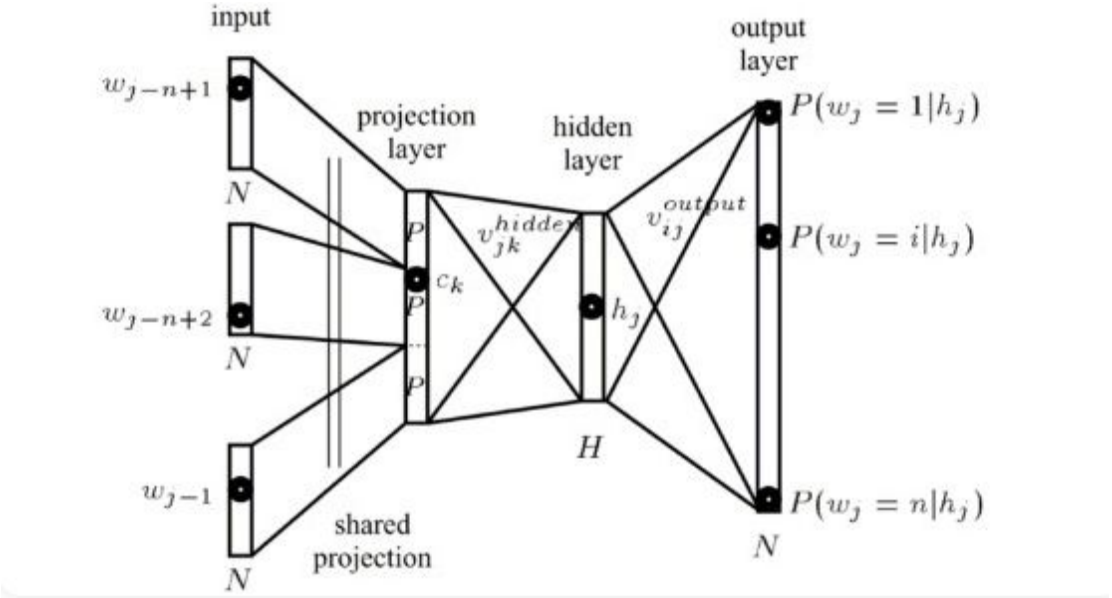

Figure 1: A Neural language model (Bengio et al., 2006)

This all lead to the eventual popularization of word embeddings in 2013, when Milolov et al. produced their paper "Efficient estimation of word representations in vector space" [2], along with the creation of *word2vec*. The primary difference

between word2vec and the proposed neural language model in Figure 1 is its computational complexity. This would explain why it took until 2013 for this jump in word embeddings popularity, as the increase in cheap computational power made the creation of models such as Word2Vec and GloVe. Partially, the reason behind word2vec's popularity was Mikolov et al. recommended two architectures for learning word embeddings that cut computational costs, Continuous bag-of-words (CBOW) and Skip-Gram.

CBOW in unlike a language model that only bases its prediction on past words, as those models are assed on predicting the next word in the corpus. This model uses both *n* words before and after the target word to predict the output. This can be seen in Figure 2.

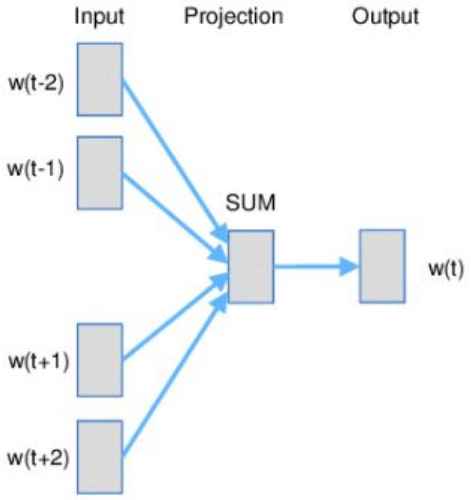

Figure 2: Continuous bag-of-words (Mikolov et al. 2013)

Rather than receiving just the previous words in the model, a section of words *n* around the target word are used. The second architecture purposed; Skip-gram, uses the logic of CBOW in reverse. Skip-gram uses the center word to predict the closest surrounding words. In actuality, it sums the log probabilities of the surrounding *n* words to the left and right of the target word.

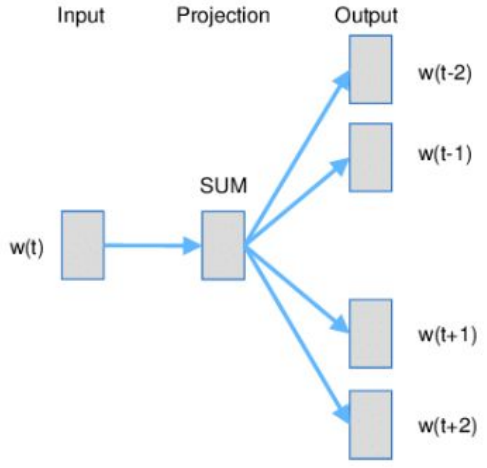

Figure 2: Continuous bag-of-words (Mikolov et al. 2013)

In a later paper within the same year [5], Mikolov et al. manage to improve greatly on the speed and accuracy of these models using additional strategies.

2.3 Lexicon Building

In general, building a lexicon for a particular concept requires significant human effort, and only a small number of human-generated concepts have been available, with usually a limited number of keywords contained in each. We now live in a world that produced more textual data daily than ever before. It is impossible to comfortably analyze any domain of text, be it medicine, politics, or finance. However, the recent popularization of word embeddings such as Word2Vec as well as other automatic text analysis methods, have led to a huge increase in sentiment analysis, documentation summarization and probabilistic topic modeling.

These methods, along with the increasing availability of digital collections of historical literature provided Leavy et al. the motivation to produce Curatr: A Platform for Semantic Analysis and Curation of Historical Literary Texts [1]. In this paper, an online platform called *Curatr* is presented. Its purpose is for "the exploration and curation of literature with machine learning-supported semantic search, designed within the context of digital humanities scholarship."

Having been trained on a large corpus of over 35000 English language digital texts from the 18th and 19th century, *Curatr* provides a text mining workflow that combines neural word embeddings, with expert domain knowledge to generate thematic lexicons of relevant topics to researchers. As mentioned, this platform makes use of word embedding models, specifically *word2vec*, over more complex language models due to the lack of structure in the corpus used to train the model, as well as OCR errors introduced in digitization of the text. To specify, the exact model used on this platform is a 100-dimensional Continuous Bag-of-Words (CBOW) *word2vec* model.

From this model, the top 20 most similar words to the given seed word are given as recommendations to the current lexicon. *Curatr,* contains a 'human in the loop' aspect, as the user must choose which of the recommended words are added to the conceptual lexicon, ensuring that the generated lexicon is informed by the domain of the user. Multiple iterations of this search leads to lexicon refinement, which accumulates in finalized lexicons used as a basis for volume retrieval from the indexed library corpus.

Chapter 3: **Progress to Date**

3.1 Data Considerations

3.2 An Outline of My Approach

3.3.1 Creation of initial Word Embedding Model

3.3.2 Comparison of Various Models

3.3.3 Visualisations of Embedding Models

Chapter 4: **Future Work**

4.1 Project Work Plan

This section will discuss my plans on future work for this project. A number of tasks have already been selected in an aim to expand my project from the work I have completed thus far. Firstly a look to expand the number of visualisations present in my work, giving a clearer comparison of different word embedding models. Having completed this, focus will change to looking for the best method of combining the output lists given from different models on the same target word to create a combined ranked list to be used by lexicon recommender in the future. Finally, I will begin working on the implementation of an interactive web interface for lexicon building. The web apps aim is to allow users to explore interactive visualisations of the different embedding models as well as accept or reject recommended words produced by the lexicon.

Before beginning expansion of my project, I hope to increase the amount of datasets being used, and potentially using new embedding algorithms such a GloVe. I feel a larger number of models being compared, and ultimately joined together will produce more appealing word recommendations.

4.1.1 Visualisation of Word Embedding Models

Having produced some visualisations of my current work already, a look to further the visual aspect of this project with an increased focus on comparison of competing word embedding models created from a variety of datasets. Highlighting the differences between two trained models from seemingly similar datasets could produce many interesting results, particularly when in relation to supposedly unbiased rival news organisations in the United States, such as a comparison between a model trained from a collection of Fox news articles and a model trained from CNN articles.

For visualising embeddings, I have several different options to choose from, including t-SNE, UMAP and TensorFlow Projector.

4.1.2 Combining of Ranked lists

Having created a number of word embedding models, comparing and evaluating these models with already available pre trained models and finding which models work best, we aim to create lexicons using a combination of the most effective models. To achieve this, I will be looking at a number of methods for combining ranked list. A method of interest in completing this task is Rank

aggregation (RA). RA is the process of combining ranked lists into a single ranking, with these methods being broken into three categories: distributional-based, heuristic and stochastic optimization. Li et al. [5], looks at these methods, characterises various types of lists, and evaluates the performance of said methods in a paper which will be a keystone to any future work I do with ranked aggregation.

4.1.3 Development of Interactive Web Interface

One of the primary aspects of this project is the implementation of a user friendly web based tool for the construction of word lexicons in an interactive and efficient manner. The ability of the user to accept or reject words produced by the model is key, to ensure that the lexicon is in the domain of the user. Additionally, a focus on including interactive visualisations for the user to explore the embedding spaces of the models they are using is required. This increased functionality will improve both user understanding of the lexicon builder, and allow for the use of different models to produce lexicons upon the users selection.

4.2 Evaluation of Web App

Evaluation is a key aspect of any produced work, which is why an aim for this project is to design and conduct a user study to assess the usefulness of the web interface and the lexicons it produces. The size and particulars of this study are yet to be determined, as the creation of the web app takes precedence, but will be required to receive any meaningful feedback on the app and it's embedding models.

4.3 Gantt Chart

Below, in Figure X, depicts a Gantt chart of the future work plan for this project. An initial start date is estimated to be on the 20th of December, finishing on the XXX.

Chapter 5: **Summary and conclusions**

References

1. Leavy, S., Meaney, G., Wade, K., Greene, D. Curatr: A Platform for Semantic Analysis and Curation of Historical Literary Texts. Proc. 13th International Conference on Metadata and Semantics Research (2019).
2. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
3. Bengio, Y., Ducharme, R., Vincent, P., & Janvin, C. (2003). A Neural Probabilistic Language Model. The Journal of Machine Learning Research, 3, 1137–1155. http://doi.org/10.1162/153244303322533223
4. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. NIPS, 1–9.
5. Xue Li, Xinlei Wang, Guanghua Xiao (2017) A comparative study of rank aggregation methods for partial and top ranked lists in genomic applications. *Briefings in Bioinformatics*, Volume 20, Issue 1, January 2019, Pages 178–189, https://doi.org/10.1093/bib/bbx101