LICENTIATE THESIS

Word Vector Representations using Shallow Neural Networks

Oluwatosin Adewumi

Machine Learning

LULEÅ UNIVERSITY OF TECHNOLOGY

Word Vector Representations using Shallow Neural Networks

Tosin Adewumi

Department of Computer Science, Electrical and Space Engineering,
Luleå University of Technology,
Luleå, Sweden

Supervisors:

Marcus Liwicki, Foteini Liwicki

To my family and friends.

Abstract

This work highlights some important factors for consideration when developing word vector representations and data-driven conversational systems. The neural network methods for creating word embeddings have gained more prominence than their older, count-based counterparts. However, there are still challenges, such as prolonged training time and the need for more data, especially with deep neural networks. Shallow neural networks with lesser depth appear to have the advantage of less complexity, however, they also face challenges, such as sub-optimal combination of hyper-parameters which produce sub-optimal models.

This work, therefore, investigates the following research questions: "How importantly do hyper-parameters influence word embeddings' performance?" and "What factors are important for developing ethical and robust conversational systems?" In answering the questions, various experiments were conducted using different datasets in different studies. The first study investigates, empirically, various hyper-parameter combinations for creating word vectors and their impact on a few Natural Language Processing (NLP) downstream tasks: Named Entity Recognition (NER) and Sentiment Analysis (SA). The study shows that optimal performance of embeddings for downstream NLP tasks depends on the task at hand. It also shows that certain combinations give strong performance across the tasks chosen for the study. Furthermore, it shows that reasonably smaller corpora are sufficient or even produce better models in some cases and take less time to train and load. This is important, especially now that environmental considerations play a prominent role in ethical research.

Subsequent studies build on the findings of the first and explore the hyper-parameter combinations for Swedish and English embeddings for the downstream NER task. The second study presents the new Swedish analogy test set for evaluation of Swedish embeddings. Furthermore, it shows that character n-grams are useful for Swedish, a morphologically rich language. The third study shows that broad coverage of topics in a corpus appears to be important to produce better embeddings and that noise may be helpful in certain instances, though they are generally harmful. Hence, a relatively smaller corpus can show better performance than a larger one, as demonstrated in the work with the smaller Swedish Wikipedia corpus against the Swedish Gigaword.

The argument is made, in the final study (in answering the second question) from the point of view of the philosophy of science, that the near-elimination of the presence of unwanted bias in training data and the use of fora like the peer-review, conferences, and journals to provide the necessary avenues for criticism and feedback are instrumental for the development of ethical and robust conversational systems.

Contents

Acknowledgments

The research work in this thesis was carried out at Luleå University of Technology within the Machine Learning Group of the Embedded Intelligent Systems Lab (EISLAB) of the Department of Computer Science, Electrical and Space Engineering.

Hence, my profound gratitude goes to the head of the Machine Learning Group: Professor Marcus Liwicki - an exemplary leader. His vision, humility and guidance have made a difference in my life. Secondly, I'm very grateful for the care and meticulous supervision of my assistant supervisor: Foteini Liwicki. The opportunities both have given me for my PhD studies have been considerable. The prompt support and attention I have received from people like Professor Jonas Ekman (the head of the department), Professor Ulf Bodin, Petter Kyösti (the head of EISLAB) and the administrative staff have been significant and I'm grateful. There are more seniors I could mention here but time and space will not permit me, including the many course instructors I had the privilege of knowing. I say thank you all.

It's impossible to forget the immense support of my family and friends halfway through this journey, whether those in Nigeria or here in Sweden. The critique, suggestions, encouragement and laughter from my colleagues, turned family, have been priceless. There are those whose hugs made my day, those whose mentor relationship inspired me and those who pleasantly "disrupted" my routine for the better, especially (in no particular order) Nosheen Abid, Saleha Javed, Maryam Pahlavan, Pedro Alonso, Sana Al-Azzawi and all the members of the Machine Learning Group. Again, for brevity, I have kept the list short. Everyone I have met, in one way or the other, have been a blessing to me and I say thank you all. Finally, there would be no me (or this work) without the All in all; I'm grateful.

Luleå, May 2021
Tosin Adewumi

Part I

Chapter 1

Introduction

"I'm pretty good with talking to girls if I have an introduction."

Bryan Greenberg

Languages are powerful means of communication and their level of development can reveal the extent of development of a given community or civilization. Natural languages have been "bequeathed" to conversational systems (or chatbots) by humans. One types on a keypad or speaks words through a channel to a chatbot and expects a response in the natural language that one understands. Underlying the communication between humans and conversational systems are technologies and algorithms developed over the years in NLP [1]. Of course, computer programs or machines do not have the natural language abilities that humans have and they have to be designed in such a way that they can be of relevance in communicating with humans.

One of the relevant technologies in NLP is word embeddings (or word vectors). They are numeric, structured representations of words in a vocabulary [2]. There have been efforts to move away from the high-dimensional and sparse word representation inherent with the bag-of-words (BoW) method. Low-dimensional, distributed embeddings provide more compact and efficient representations [3]. Deep neural networks, such as the Bidirectional Encoder Representations from Transformers (BERT, with WordPiece embeddings) [4] and Generative Pre-trained Transformer (GPT2) [5], have taken advantage of the efficiency of such by combining pre-training, that involves learning word vectors, and supervised fine-tuning. In such deep neural network (NN), usually, the embedding process is done simultaneously with the rest of the model development [4]. With shallow NN, the embeddings can be created separately and may be supplied to another network for downstream tasks [6]. Whatever the type of NN, the goal is to generalize from training data by discovering similarities between words [2].

Various approaches have been introduced to achieve low-dimensional, distributed embeddings. These include GloVe [6], word2vec [7] and fastText [8], among others. There are a number of factors or properties for consideration for any given NN, such as the depth of the network and the number of neurons in each layer. The performance of any

NN is dependent on these properties, called hyper-parameters, set by the developer or designer. As expected, the more the number of layers or neurons to a given network, the higher the complexity of the network and the higher the number of hyper-parameter combinations that can be set in the network [7]. Equation 1.1 gives the training complexity, where E, T and Q are the training epochs, the number of words in the training data and additional architectural factors, respectively.

$$O = E * T * Q \tag{1.1}$$

Equation 1.2 describes a representation of the conditional probability of the next word given the previous ones. This is a statistical model of language [2].

$$P(w_1^T) = \prod_{t=1}^{T} P(w_t | w_1^{t-1}) \tag{1.2}$$

There are different methods that can be used when exploring the combination of hyper-parameters during training of an NN. Some of these methods include grid search and Bayesian optimization [9]. Both are explored at various instances in this work. Grid search is applied to the models of word2vec and fastText while Bayesian optimization is applied during the NER task for both English and Swedish. In this thesis work, the author investigates the importance of hyper-parameter combination for generating quality word embeddings for NLP downstream tasks. This investigation is carried out for the following natural languages: English, Swedish and Yorùbá, as presented in the appended papers.

1.1 Research Problems Formulation

Given some of the challenges identified earlier and those with creating word embeddings, the following research questions were formulated in order to be addressed.

1. How importantly do hyper-parameters influence word embeddings' performance?
2. What factors are important for developing ethical and robust conversational systems?

1.1.1 Delimitation

The scope of this licentiate thesis includes investigation of the combination of a limited number of hyper-parameters, covering three natural languages, and a few NLP downstream tasks. The three natural languages discussed, in varying details, are English, Swedish and Yorùbá. Furthermore, the NNs experimented with are the continuous Skip-gram and continuous Bag-of-Words (CBoW) architectures of the word2vec and fastText models.

This work does not cover all combinations of hyper-parameters possible for a given NN. It is not very practical to cover all possible hyper-parameter combinations for the

NN, as the combination geometrically increases with additional hyper-parameters. Also, this work does not experiment with all shallow neural networks available nor does it cover all NLP downstream tasks.

Finally, the discussion in this work about conversational systems only prepares the ground for ongoing and future work. It mainly highlights the identified factors, which are important to ethical and robust conversational systems from the point of view of the philosophy of science.

1.2 Thesis Outline

This thesis is divided into two main parts: Part I, which includes the introductory chapters (including the kappa) and Part II, which contains five appended papers. Chapter 1 introduces the key concepts and sets the stage for why the particular research questions were formulated. Chapter 2 covers some of the concepts, like word vectors, shallow neural networks and performance metrics, in more detail in a literature review. Chapter 3 describes briefly the experiments conducted, the methodology used and the overview of results. Chapter 4 presents the five papers of this thesis by introducing their titles, abstracts and author contributions. The final chapter, Chapter 5, concludes Part I of the thesis with concluding summaries and motivation for future work.

Part II contains two papers under review and three published ones, including two peer-reviewed conference/workshop papers and one journal paper. The two papers under review are titled "Word2Vec: Optimal Hyper-Parameters and Their Impact on NLP Downstream Tasks" and "Exploring Swedish & English fastText Embeddings for NER with the Transformer". The two conference papers are "Corpora Compared: The Case of the Swedish Gigaword & Wikipedia Corpora" and "The Challenge of Diacritics in Yorùbá Embeddings" presented at the Swedish Language Technology Conference (SLTC) 2020 and ML4D NeurIPS 2020, respectively. The journal paper is titled "Conversational Systems in Machine Learning from the Point of View of the Philosophy of Science — Using Alime Chat and Related Studies" and was published by the journal *Philosophies*.

Chapter 2

Literature Review

"Torture the data and it will confess to anything."

Ronald Coase

Representation of words in NLP began with simple approaches like the one-hot encoding and bag-of-words, which have the inherent limitation of indifference to word order [3]. The n-gram model, a statistical language model, is another example of such [7]. They are also incapable of representing idiomatic phrases [3]. Idioms are Multi-Word Expression (MWE) that have unrelated meaning to those of the individual words that make them up [10]. They pose challenges in NLP tasks such as Machine Translation (MT) and metonymy resolution [11]. The disadvantages (including, for example, the curse of dimensionality) of such very simple methods were quickly apparent and researchers sought new ways of representing words and sub-words.

2.1 Word Vectors

Using low-dimensional, distributed embeddings give more efficient representations [3]. Tables 2.1, 2.2 and 2.3 compare the ways word vectors (using the example sentence "The cat sat on the mat, next to the mouse.") may be represented using one-hot encoding, bag of words and low-dimensional, distributed representation, respectively. In some NLP tasks, pre-processing will involve lowering all cases, the removal of punctuation and frequent, but less informative, words like 'the'.

The introduction of models such as the continuous Skip-gram and CBoW [7, 3] brought improvements to word vector representations. The architectural diagram for both are shown in figure 2.1. The Skip-gram model objective is predicting context words by learning vector representations. This is expressed in Equation 2.1[3], where context size is given by c. On the other hand, the CBoW has the objective of predicting the center word [7]. The hierarchical softmax and negative sampling are alternative functions that can be applied to either of the architectures in word2vec. Subsampling of frequent words is used to counter imbalance in rare and frequent words.

As an example of the advantage of low-dimensional, distributed embeddings, the

Table 2.1: Example of One-hot Encoding

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
the	1	0	0	0	0	0	0	0
cat	0	1	0	0	0	0	0	0
sat	0	0	1	0	0	0	0	0
on	0	0	0	1	0	0	0	0
mat	0	0	0	0	1	0	0	0
next	0	0	0	0	0	1	0	0
to	0	0	0	0	0	0	1	0
mouse	0	0	0	0	0	0	0	1

Table 2.2: Example of Bag-of-Words

Term:	the	cat	sat	on	mat	next	to	mouse
Frequency:	3	1	1	1	1	1	1	1

Table 2.3: Example of Low-dimensional, Distributed Representation

	1	**2**	**3**	**4**
the	0.023	0.011	-0.013	0.201
cat	0.11	-0.23	0.132	-0.221
sat	0.312	0.033	0.078	0.091
on	-0.165	0.099	0.076	0.045
mat	0.088	0.109	0.076	0.023
next	0.156	-0.066	0.231	0.002
to	0.002	0.014	-0.055	0.311
mouse	0.113	-0.33	0.152	-0.422

challenge of the curse of dimensionality is somewhat mitigated [2]. Different aspects of a word are represented in these feature vectors and the number of features is far smaller compared to the vocabulary size. In this approach, similar words, in terms of semantics and syntax, produce similar feature vectors [2].

$$\frac{1}{T}\sum_{i=1}^{T}\sum_{-c\leq j\leq c, j\neq 0} log\ p(w_{t+j}|w_t) \tag{2.1}$$

2.2 Shallow Neural Networks

An artificial neural network (ANN) contains connected neurons at different depths. The NN is termed shallow when the depth is only a few layers (say, two or three). The NN is used to predict the next word, based on previous words in the context [2]. The n-gram method is different and achieves less significant results when compared with the NN method [2]. Improving the results of NLP tasks using NN can involve the introduction of

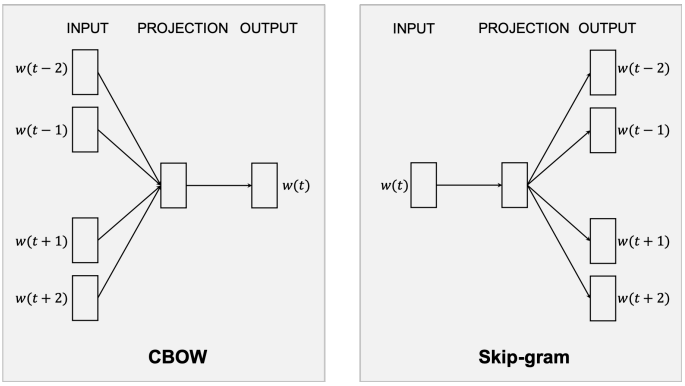

Figure 2.1: The CBoW and continuopus Skip-gram model architectures [7]

a-priori knowledge [2]. Such knowledge may include semantic information from WordNet and grammatical information from parts-of-speech (PoS).

[3] found out that the choice of hyper-parameters is task-specific, as different tasks perform well under different configurations. The following were regarded as the most important in their work: model architecture, the training window, subsampling rate and the dimension size of the vector. Furthermore, [12] revealed that choices of hyper-parameters have major impact on the performance of models.

In Latent Semantic Indexing (or Analysis), feature vectors are learned based on the probability of co-occurence in the same documents [2, 13]. The technique estimates continuous representations of words using singular-value decomposition [7, 2]. This is unlike the continuous representations learned by NN methods. [6] introduced GloVE, another method for low-dimensional, distributed representations of words. It is a global log-bilinear regression model that uses matrix factorization and local context window. Furthermore, it is a statistical model that trains on word-word co-occurrence matrix in a corpus.

fastText, introduced by [8], brought gains to the original methods in word2vec, extending the same architectures. It sometimes achieved accuracy performance at par with deep learning classifiers while much faster for training and evaluation. Subword vectors in fastText addressed the morphology of words by treating each word as the sum of a bag of character n-grams [14]. The model addresses out-of-vocabulary words by building vectors for words that do not appear in the training data [14].

2.3 Data

Clean data (with as little noise as possible) is essential in training NNs, just as the size of the data is also essential [14]. The NN pipeline usually uses three splits of data: the training set, the validation (or development) set and the test set [15]. The test set is

used to determine how well the model generalizes after training while the validation set is used to determine the best choice of model, weight decay and other hyper-parameters during training. The training data is fed to the neural network in order to maximize its log-likelihood when the parameters of the probability function are iteratively tuned [2]. The analogy test set is used as a reasoning task in evaluating word embeddings [7, 3]. Different versions in different languages have evolved in this regard [16, 17]. Examples of training data used in generating word embeddings include Google News [7], Common Crawl, Gigaword [18, 6] and Wikipedia [14].

2.4 NLP Tasks

Downstream NLP tasks are what finally matter to users of NLP systems [19]. There are many of such tasks [19, 20] and some of them are listed below .

- Named Entity Recognition (NER) - this involves the classification of specific entities.
- Sentiment Analysis (SA) - this involves classification of sentences/text according to sentiments.
- Content Determination - this involves determining the information to be communicated.
- Text Structuring - this involves determining the order of presentation of texts.
- Sentence Aggregation - this involves grouping of related messages.
- Lexicalization - this involves determining words or phrases for expression.
- Referring Expression Generation - this involves selecting words to identify domain entities.
- Linguistic Realisation - this involves generating the right morphological forms.
- Text Summarization - this involves summarizing relevant points within a large text.
- Machine Translation - this involves translating text from one language to a second language.

2.5 Performance Metrics

Accuracy result from the analogy reasoning task is used as an evaluation metric. The task is based on using cosine distance to find a vector that is closest to the true value of the vector arithmetic involved in a pair of two related words [3]. This is one of the intrinsic evaluation methods available [21, 22]. WordSim-353 is another common intrinsic

evaluation task [23, 6]. Although intrinsic evaluation methods like the analogy reasoning task (or simply, word analogy task [14]) have been shown to have weaknesses [21], they are still used as proxies for ascertaining the possible performance of embeddings on downstream NLP tasks [24, 22, 25]. Spearman correlation is also used for evaluation. [14] computed Spearman correlation between human judgement and the cosine similarity between representations.

Extrinsic evaluation methods focus on the usefulness of models with regards to downstream NLP tasks, such as Named Entity Recognition (NER) [22]. Such evaluations are carried out when embeddings are employed in NNs (involving architectures like the LSTM) for specific tasks [26, 27, 28, 29]. The common metrics for extrinsic evaluation include accuracy, precision, recall and the F1 score [19]. They are represented mathematically in Equations 2.2, 2.3, 2.4 and 2.5, respectively, using the concepts of true positive (TP, the number of items correctly classified as positive instances), true negative (TN, the number of items correctly classified as negative instances), false negative (FN, the number of items incorrectly classified as negative instances) and false positive (FP, the number of items incorrectly classified as positive instances).

$$\frac{TP + TN}{TP + TN + FP + FN} \tag{2.2}$$

$$\frac{TP}{TP + FP} \tag{2.3}$$

$$\frac{TP}{TP + FN} \tag{2.4}$$

$$\frac{2TP}{2TP + FP + FN} \tag{2.5}$$

The F1 score is the harmonic mean of both the precision and recall [30]. These are the metrics used in this work.

Chapter 3

Experiments

"The true method of knowledge is experiment."

William Blake

This chapter summarizes common areas of experiments among the papers. Details of each experiment are contained in the respective papers, including models generated and the data involved. Papers A to D involve experiments. Experiments are not applicable to Paper E, as it presents an argument from a philosophical point of view.

Experiments in paper A involve generating word2vec embeddings, using different hyper-parameter combinations, and deploying them in two downstream tasks: NER and SA. Experiments in paper B involve generating Swedish and English fastText embeddings, using some established hyper-parameter combinations from paper A, and deploying them in NER tasks for both languages by using the Transformer architecture. Experiments in paper C involve intrinsic evaluation of two differently-sized Swedish corpora, by using the new Swedish analogy test set that the authors introduced. This was done by generating fastText embeddings with both corpora. Paper D involves generating fastText embeddings for two versions of the written Yorùbá language for evaluation of the effect of diacritics (tonal marks) on intrinsic performance, which was measured by the newly introduced Yorùbá analogy test set, in addtion to Yorùbá version of WordSim-353.

3.1 Methodology & Implementation

Similar methodology was employed in all relevant aspects of the experiments of all the papers. They were run on a shared cluster running the Ubuntu operating system. Gensim Python library program was used to evaluate all models against their corresponding analogy test sets. Relevant data pre-processing, such as removal of punctuation marks and lowering of cases, was performed before training. Running each embedding training multiple times to obtain averages would have been ideal but because of the limited time available, a work-around was adopted, which was to run a few random models twice to ascertain if there were major differences per model. Since there were no major differences, this method was adopted for building embeddings in papers A and B, as they involved

larger training data for longer periods. This is especially the case for paper A, which used the Python Gensim library for word2vec models, which is slower for being an interpreted language [31].

Pytorch deep learning framework was used for the downstream tasks. In all cases for the downstream tasks, the dataset was shuffled before training and split in the ratio 70:15:15 for training, dev (or validation) and test sets. For each task, experiments for each embedding were conducted several times and an average value was calculated. The long short term memory network (LSTM) and the BiLSTM were used for the downstream tasks in paper A.

In papers B, C and D, all pre-trained models were generated using the original C++ implementation of fastText. Some of the default hyper-parameter settings (e.g. the initial learning rate of 0.05) were retained [14]. The English and Swedish training data were pre-processed using the recommended script [32]. The Transformer Encoder architecture in PyTorch was utilized for the downstream NER task in paper B and three hyper-parameters were tuned using SigOpt (a Bayesian hyper-parameter optimization tool).

3.2 Performance Metrics

For intrinsic evaluation, analogy test sets for the corresponding languages were utilized. The WordSim-353 was also used for the English embeddings. The WordSim result output file from the Gensim Python program always has more than one value reported, including the Spearman correlation. The first value (reported as WordSim score1) and the Spearman correlation are always reported in the relevant papers concerned. An example output for the embedding by [32] is given below:

> *((0.6853162842820049, 2.826381331182216e-50), SpearmanrResult(correlation=0.70236817646248, pvalue=9.157442621319373e-54), 0.0)*

For the extrinsic evaluation, F1 scores, precision and recall are reported for the downstream tasks concerned. Accuracy is additionally reported for SA in paper A.

3.3 Results Overview

The following sub-sections present results from the papers in a very brief format.

3.3.1 Paper A: Word2Vec: Optimal Hyper-Parameters and Their Impact on NLP Downstream Tasks

Table 3.1 summarizes key results from the intrinsic evaluations[1]. Table 3.2 reveals the training time (in hours) and average embedding loading time (in seconds), representative

[1]The results are to 3 decimal places

Table 3.1: Scores for 300 dimensions for 10 epochs for SW, BW & 100B corpora.

	w8s1h1	**w8s0h1**	**w8s0h0**	**w8s1h0**	**w4s1h1**	**w4s0h1**	**w4s0h0**	**w4s1h0**
Simple Wiki								
Analogy	0.461	0.269	**0.502**	0.439	0.446	0.243	0.478	0.407
WordSim score1	0.636	0.611	0.654	**0.655**	0.635	0.608	0.620	0.635
Spearman	0.670	0.648	0.667	**0.695**	0.668	0.648	0.629	0.682
Billion Word								
Analogy	0.587	0.376	0.638	0.681	0.556	0.363	0.629	**0.684**
WordSim score1	0.614	0.511	0.599	**0.644**	0.593	0.508	0.597	0.635
Spearman	0.653	0.535	0.618	**0.681**	0.629	0.527	0.615	0.677
Google News - 100B (s1h0)								
Analogy: 0.740			WordSim score1: 0.624			Spearman: 0.659		
Key: w = window size; s1 = Skip-gram; s0 = CBoW; h1 = hierarchical softmax; h0 = negative sampling								

of the various models used. Tables 3.3 and 3.4 summarize key results for the extrinsic evaluations. The embedding by [7] beats our best models in only analogy score (even for Simple Wiki (SW)) despite using a much bigger corpus of 3,000,000 vocabulary size and 100 billion words while SW had vocabulary size of 367,811 and is 711MB. It is very likely our analogy scores will improve when we use a much larger corpus, as can be observed from table 3, which involves just one billion words.

Significance tests using bootstrap, based on [33], on the results of the differences in the means are reported for the downstream tasks and we conclude the difference for NER was likely due to chance and fail to reject the null hypothesis but for SA the difference is unlikely due to chance so we reject the null hypothesis.

Table 3.2: Training & embedding loading time for w8s1h0, w8s1h1 & 100B

Model	**Training (hours)**	**Loading Time (s)**
SW w8s1h0	5.44	1.93
BW w8s1h1	27.22	4.89
GoogleNews (100B)	-	97.73

Table 3.3: NER Dev and Test sets Mean Results

Metric	**Default**	**100B**	**w8 s0 h0**	**w8 s1 h0**	**BW w4 s1 h0**
	Dev, Test	Dev, Test	Dev, Test	Dev, Test	Dev, Test
F1	0.661, 0.661	**0.679**, 0.676	0.668, 0.669	0.583, 0.676	**0.679, 0.677**
Precision	0.609, 0.608	**0.646, 0.642**	0.636, 0.637	0.553, 0.642	0.644, **0.642**
Recall	0.723, **0.724**	0.716, 0.714	0.704, 0.706	0.618, 0.715	0.717, 0.717

Table 3.4: Sentiment Analysis Dev and Test sets Mean Results

Metric	Default	100B	w8 s0 h0	w8 s1 h0	BW w4 s1 h0
	Dev, Test	Dev, Test	Dev, Test	Dev, Test	Dev, Test
F1	**0.810, 0.805**	0.384, 0.386	0.798, 0.799	0.548, 0.553	0.498, 0.390
Precision	0.805, 0.795	0.6, 0.603	**0.814, 0.811**	0.510, 0.524	0.535, 0.533
Recall	**0.818, 0.816**	0.303, 0.303	0.788, 0.792	0.717, 0.723	0.592, 0.386
Accuracy	**0.807, 0.804**	0.549, 0.55	0.801, 0.802	0.519, 0.522	0.519, 0.517

3.3.2 Paper B: Exploring Swedish & English fastText Embeddings for NER with the Transformer

Intrinsic results for the pre-trained models are given in table 3.5. An important trend that can be observed is the higher scores for Skipgram-negative sampling in all the cases (English & Swedish), except one. This appears to confirm previous research [7, 34]. The English word2vec embedding by [7] is represented as *'GN'* in the table while that by [32], trained on the Common Crawl & Wikipedia, are represented by *'Gr'*. Tables 3.6 and 3.7 present the results of the NER task for the selected English & Swedish embeddings, respectively. The Swedish subword embeddings outperform the word2vec ones, implying that character n-grams are useful for Swedish. Significance tests using bootstrap, based on [33], on the results of the differences in the means are reported for the downstream task and we conclude the difference is unlikely due to chance for English but for Swedish the difference is likely due to chance.

Table 3.5: Intrinsic Scores - English & Swedish (highest score/row in bold)

	Skipgram (s1)				**CBoW (s0)**					
	H. S. (h1)		**N. S. (h0)**		**H. S. (h1)**		**N. S. (h0)**		**Gr**	**GN**
window (w)	**4**	**8**	**4**	**8**	**4**	**8**	**4**	**8**		
Subword %										
Analogy	62.6	58.8	74.4	69.8	67.2	68.7	71.6	71	**82.6**	
WordSim score1	64.8	66.3	69.9	**70**	62.6	66.2	47.3	51.1	68.5	
Spearman	67.6	69.4	**74.3**	73.6	65.3	70.3	45.3	49.5	70.2	
Word2Vec %										
Analogy	61.3	58.3	73.5	70.4	59.7	61.9	**76.2**	75.4		74
WordSim score1	66.3	67.3	69.6	**70.1**	64.1	66.7	65.4	67.5		62.4
Spearman	70	70.9	74.5	**74.7**	68.2	71.2	66.9	69.4		65.9
Swedish										
Subword %	45.05	39.99	53.53	53.36	26.5	23.93	36.79	35.89	**60.9**	
Word2Vec %	45.53	41.21	**58.25**	57.30	28.02	28.04	52.81	55.64		

Table 3.6: English NER Mean Scores

					Word2Vec (W)						Subword			
Metric	**Default**		**Gr**		**w8s0h0**		**w4s0h0**		**w4s1h0**		**w4s0h0**		**w8s1h1**	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test
F1	0.719	**0.723**	0.588	0.6602	0.719	0.720	0.715	0.716	0.714	0.716	0.695	0.668	0.592	0.684
Precision	0.685	0.69	0.564	0.634	0.689	**0.691**	0.686	0.688	0.684	0.686	0.664	0.64	0.567	0.656
Recall	0.756	**0.759**	0.615	0.689	0.751	0.752	0.747	0.747	0.748	0.748	0.729	0.7	0.62	0.713

Table 3.7: Swedish NER Mean Scores

					Word2Vec (W)				Subword					
Metric	**Default**		**Gr**		**w4s1h0**		**w8s0h1**		**w4s1h1**		**w4s1h0**		**w8s0h1**	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test
F1	0.487	**0.675**	0.441	0.568	0.574	0.344	0.477	0.429	0.507	0.649	0.492	0.591	0.486	0.623
Precision	0.51	0.745	0.682	**0.856**	0.704	0.549	0.626	0.669	0.647	0.821	0.658	0.752	0.626	0.802
Recall	0.471	**0.633**	0.331	0.44	0.489	0.265	0.398	0.325	0.420	0.543	0.398	0.5	0.402	0.524

3.3.3 Paper C: Corpora Compared: The Case of the Swedish Gigaword & Wikipedia Corpora

Table 3.8 gives mean analogy scores for learning rate (LR) of 0.05 of the embeddings for the two corpora and table 3.9 for LR of 0.01. From table 3.8, the highest score is achieved by the Wikipedia word2vec embedding with 60.38%. Also, the Wikipedia embeddings have higher analogy scores than their Gigaword counterparts. Apparently, the general better performance observed between the embeddings of the two corpora is because of the wider domain coverage of the Wikipedia corpus and the small noise in the Wikipedia corpus, caused by the pre-processing script by [32].

Table 3.8: Mean Analogy Scores for Swedish Gigaword & Wikipedia Corpora with LR=0.05

	Skipgram (s1)				**CBoW (s0)**			
	H. S. (h1)		**N. S. (h0)**		**H. S. (h1)**		**N. S. (h0)**	
window (w)	**4**	**8**	**4**	**8**	**4**	**8**	**4**	**8**
Word2Vec %								
Wikipedia	47.02	44.09	**60.38**	60.38	29.09	30.09	54.39	56.81
Gigaword	40.26	44.23	**55.79**	55.21	26.23	27.82	55.2	55.81
Subword %								
Wikipedia	46.65	45.8	**56.51**	56.36	28.07	24.95	38.26	35.92
Gigaword	41.37	44.7	**58.31**	56.28	2.59	-	46.81	46.39

Table 3.9: Analogy Scores for Swedish Gigaword & Wikipedia Corpora with LR=0.01

	Skipgram (s1)				**CBoW (s0)**			
	H. S. (h1)		**N. S. (h0)**		**H. S. (h1)**		**N. S. (h0)**	
window (w)	**4**	**8**	**4**	**8**	**4**	**8**	**4**	**8**
Word2Vec %								
Wikipedia	48.92	49.01	51.71	**53.48**	32.36	33.92	47.05	49.76
Gigaword	39.12	43.06	48.32	**49.96**	28.89	31.19	44.91	48.02
Subword %								
Wikipedia	45.16	**46.82**	35.91	43.26	22.36	21.1	14.31	14.45
Gigaword	39.13	43.65	45.51	**49.1**	31.67	35.07	28.34	28.38

3.3.4 Paper D: The Challenge of Diacritics in Yorùbá Embeddings

Tables 3.10 and 3.11 show results from the experiments. Average results for embeddings from the 3 training datasets and the embedding by [32] are tabulated: Wiki, U_Wiki, C3 & CC, representing embeddings from the cleaned Wikipedia dump, its undiacritized (normalized) version and the other two sources, including the embedding by [32]. It can be observed from table 3.10 that the cleaned Wiki embedding has lower scores than the C3 because of noise, despite the larger data size of the Wiki. Inspite of this noise, the exact undiacritized version (U_Wiki) outperforms C3, giving the best WordSim score1 and Spearman correlation. This seems to show diacritized data affects Yorùbá embeddings.

Table 3.10: Yorùbá word2vec embeddings intrinsic scores (%)

Data	Vocab	Analogy	WordSim score1	Spearman
Wiki	275,356	0.65	26.0	24.36
U_Wiki	269,915	**0.8**	**86.79**	**90**
C3	31,412	0.73	37.77	37.83

Table 3.11: Yorùbá subword embeddings intrinsic scores (%)

Data	Vocab	Analogy	WordSim score1	Spearman
Wiki	275,356	0	45.95	44.79
U_Wiki	269,915	0	72.65	60
C3	31,412	0.18	39.26	38.69
CC	151,125	4.87	16.02	9.66

Chapter 4

Contributions

"A year spent in artificial intelligence is enough to make one believe in God."

Alan Perlis

4.1 Paper A: Word2Vec: Optimal Hyper-Parameters and Their Impact on NLP Downstream Tasks

Title Word2Vec: Optimal Hyper-Parameters and Their Impact on NLP Downstream Tasks

Authors Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Abstract Word2Vec is a prominent model for natural language processing (NLP) tasks. Similar inspiration is found in distributed embeddings for new state-of-the-art (SotA) deep neural networks. However, wrong combination of hyper-parameters can produce poor quality vectors. The objective of this work is to empirically show that optimal combination of hyper-parameters exists and evaluate various combinations. We compare them with the released, pre-trained original word2vec model. Both intrinsic and extrinsic (downstream) evaluations, including named entity recognition (NER) and sentiment analysis (SA) were carried out. The downstream tasks reveal that the best model is usually task-specific, high analogy scores don't necessarily correlate positively with F1 scores and the same applies to the focus on data alone. Increasing vector dimension size after a point leads to poor quality or performance. If ethical considerations to save time, energy and the environment are made, then reasonably smaller corpora may do just as well or even better in some cases. Besides, using a small corpus, we obtain better WordSim scores, corresponding Spearman correlation and better downstream performances (with significance tests) compared to the original model, trained on a 100 billion-word corpus.

Personal Contributions Conceptualization and Methodology by Tosin Adewumi. Refining of Concept and Methodology by Marcus Liwicki. Experiments were run by Tosin

Adewumi. Original draft preparation by Tosin Adewumi. Review and supervision by Foteini Liwicki and Marcus Liwicki.

4.2 Paper B: Exploring Swedish & English fastText Embeddings for NER with the Transformer

Title Exploring Swedish & English fastText Embeddings for NER with the Transformer

Authors Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Abstract In this paper, our main contributions are that embeddings from relatively smaller corpora can outperform ones from larger corpora and we make the new Swedish analogy test set publicly available. To achieve a good network performance in natural language processing (NLP) downstream tasks, several factors play important roles: dataset size, the right hyper-parameters, and well-trained embeddings. We show that, with the right set of hyper-parameters, good network performance can be reached even on smaller datasets. We evaluate the embeddings at both the intrinsic and extrinsic levels. The embeddings are deployed with the Transformer in named entity recognition (NER) task and significance tests conducted. This is done for both Swedish and English. We obtain better performance in both languages on the downstream task with smaller training data, compared to recently released, Common Crawl versions and character n-grams appear useful for Swedish, a morphologically rich language.

Personal Contribution Conceptualization and Methodology by Tosin Adewumi. Refining of Concept and Methodology by Marcus Liwicki. Experiments were run by Tosin Adewumi. Original draft preparation by Tosin Adewumi. Review and supervision by Foteini Liwicki and Marcus Liwicki.

4.3 Paper C: Corpora Compared: The Case of the Swedish Gigaword & Wikipedia Corpora

Title Corpora Compared: The Case of the Swedish Gigaword & Wikipedia Corpora

Authors Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Abstract In this work, we show that the difference in performance of embeddings from differently sourced data for a given language can be due to other factors besides data size. Natural language processing (NLP) tasks usually perform better with embeddings from bigger corpora. However, broadness of the covered domain and noise can play important roles. We evaluate embeddings based on two Swedish corpora: The Gigaword and Wikipedia, in analogy (intrinsic) tests and discover that the embeddings from the Wikipedia corpus generally outperform those from the Gigaword corpus, which is a big-

ger corpus. Downstream tests will be required to have a definite evaluation.

Personal Contribution Conceptualization and Methodology by Tosin Adewumi. Refining of Concept and Methodology by Foteini Liwicki and Marcus Liwicki. Experiments were run by Tosin Adewumi. Original draft preparation by Tosin Adewumi. Review and supervision by Foteini Liwicki and Marcus Liwicki.

4.4 Paper D: The Challenge of Diacritics in Yorùbá Embeddings

Title The Challenge of Diacritics in Yorùbá Embeddings

Authors Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Abstract The major contributions of this work include the empirical establishment of a better performance for Yorùbá embeddings from undiacritized (normalized) dataset and provision of new analogy sets for evaluation. The Yorùbá language, being a tonal language, utilizes diacritics (tonal marks) in written form. We show that this affects embedding performance by creating embeddings from exactly the same Wikipedia dataset but with the second one normalized to be undiacritized. We further compare average intrinsic performance with two other work (using analogy test set & WordSim) and we obtain the best performance in WordSim and corresponding Spearman correlation.

Personal Contribution Conceptualization and Methodology by Tosin Adewumi. Experiments were run by Tosin Adewumi. Original draft preparation by Tosin Adewumi. Review and supervision by Foteini Liwicki and Marcus Liwicki.

4.5 Paper E: Conversational Systems in Machine Learning from the Point of View of the Philosophy of Science — Using Alime Chat and Related Studies

Title Conversational Systems in Machine Learning from the Point of View of the Philosophy of Science — Using Alime Chat and Related Studies

Authors Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Abstract This essay discusses current research efforts in conversational systems from the philosophy of science point of view and evaluates some conversational systems research activities from the standpoint of naturalism philosophical theory. Conversational systems or chatbots have advanced over the decades and now have become mainstream applications. They are software that users can communicate with, using natural lan-

guage. Particular attention is given to the Alime Chat conversational system, already in industrial use, and the related research. The competitive nature of systems in production is a result of different researchers and developers trying to produce new conversational systems that can outperform previous or state-of-the-art systems. Different factors affect the quality of the conversational systems produced, and how one system is assessed as being better than another is a function of objectivity and of the relevant experimental results. This essay examines the research practices from, among others, Longino's view on objectivity and Popper's stand on falsification. Furthermore, the need for qualitative and large datasets is emphasized. This is in addition to the importance of the peer-review process in scientific publishing, as a means of developing, validating, or rejecting theories, claims, or methodologies in the research community. In conclusion, open data and open scientific discussion fora should become more prominent over the mere publication-focused trend.

Personal Contribution Conceptualization and Methodology by Tosin Adewumi. Refining of Concept and Methodology by Foteini Liwicki and Marcus Liwicki. Original draft preparation by Tosin Adewumi. Review and supervision by Foteini Liwicki and Marcus Liwicki.

Chapter 5

Conclusion and Future Work

"An end is only a beginning in disguise."

Craig Lounsbrough

5.1 Conclusion

Considerable success has been made in NLP over the years with regards to word vector representations. The success has been instrumental to development in related areas like open-domain conversational systems that use deep models for generating dialogues [35, 36]. It is also the case that the complexity of neural networks increases with increasing hyper-parameters or other network factors [7]. Hence, this work set out to investigate the following research questions:

1. How importantly do hyper-parameters influence word embeddings' performance?
2. What factors are important for developing ethical and robust conversational systems?

Given the first question, paper A empirically reveals that hyper-parameters importantly influence performance of word embeddings. It shows that optimal performance of embeddings (based on hyper-parameter combinations) for downstream NLP tasks varies with the NLP tasks. However, some combinations give strong performance across the tasks chosen for the study: NER and SA. This is specifically for the tested word2vec model architectures. It also shows that high analogy scores do not always correlate positively with downstream tasks. Furthermore, an increase in embedding dimension size depreciates performance after a point. Environmental considerations give importance to certain choices of hyper-parameters, as reasonably smaller corpora suffice or even produce better models in some cases.

Paper B builds on the findings of paper A and explores the hyper-parameter combinations for Swedish and English embeddings for the downstream NER task. It presented the new Swedish analogy test set for evaluation of Swedish embeddings. The work reveals the

trend of better performance with Skipgram-negative sampling pre-trained models across the two languages. Furthermore, it shows that character n-grams are useful for Swedish, a morphologically rich language. It establishes that increasing only the training data size does not equate to better performance, as other hyper-parameters contribute to better performance.

Paper C reveals that broad coverage of topics in a corpus seems important for better embeddings and that noise, though generally harmful, may be helpful in certain instances. Hence, a relatively smaller corpus can show better performance than a larger one, as demonstrated in the work with the smaller Swedish Wikipedia corpus against the Swedish Gigaword. Paper D then shows that it appears advantageous normalizing diacritized (tonal marks) texts for NLP tasks, since they produce better intrinsic performance, generally.

Finally, the argument was put forward in paper E (in answering the second question) for factors important for developing ethical and robust conversational systems through machine learning, from the point of view of the philosophy of science. The efforts to be made in this regard include the elimination (or near-elimination) of the presence of unwanted bias or stereotypes in training data and the use of fora like the peer-review, conferences, workshops, and journals to provide the necessary avenues for criticism and feedback.

As there are limitations to the volume of work that can be carried out in a limited time, this work is also limited in scope to the investigation of the combination of a limited number of hyper-parameters, covering three natural languages (English, Swedish and Yorùbá), and a few NLP downstream tasks. Also, not all shallow NNs were experimented with but the following: the continuous Skip-gram and CBoW architectures.

5.2 Future Work

Future work will investigate Natural Language Generation (NLG) in multiple languages (English, Swedish and Yorùbá) by building data-driven, open-domain conversational systems, using deep models, based on the findings of this work. Vector representations of idioms will also be covered. As part of representing idioms, it will be required to create a fairly large dataset with multiple classes of the available ones. Currently, all surveyed idioms datasets only distinguish between literal and idiomatic expressions and do not have classes that cover the various idioms available. Therefore, the conversational systems planned for future work should be able to distinguish idiomatic expressions from literal ones during conversations. This will require adjusting an existing deep model to accomplish the task.

In addition, investigating the linguistic and mathematical features across the languages for which conversational systems will be built should be an interesting piece of work. One-dimensional and multiple-dimensional visualizations, based on metrics for evaluation, may be graphed to see if any interesting observations (such as relatedness of the languages) can be made.

References

[1] J. Weizenbaum, "Eliza—a computer program for the study of natural language communication between man and machine," *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966.

[2] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *The journal of machine learning research*, vol. 3, pp. 1137–1155, 2003.

[3] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, pp. 3111–3119, 2013.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[5] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[6] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[7] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[8] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," *arXiv preprint arXiv:1607.01759*, 2016.

[9] R. Martinez-Cantin, K. Tee, and M. McCourt, "Practical bayesian optimization in the presence of outliers," in *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics* (A. Storkey and F. Perez-Cruz, eds.), vol. 84 of *Proceedings of Machine Learning Research*, (Playa Blanca, Lanzarote, Canary Islands), p. 1722–1731, PMLR, 09–11 Apr 2018.

[10] A. Quinn and B. R. Quinn, *Figures of speech: 60 ways to turn a phrase*. Psychology Press, 1993.

[11] I. Korkontzelos, T. Zesch, F. M. Zanzotto, and C. Biemann, "Semeval-2013 task 5: Evaluating phrasal semantics," in *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pp. 39–47, 2013.

[12] O. Levy, Y. Goldberg, and I. Dagan, "Improving distributional similarity with lessons learned from word embeddings," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 211–225, 2015.

[13] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, no. 6, pp. 391–407, 1990.

[14] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[15] A. Belz and E. Reiter, "Comparing automatic and human evaluation of nlg systems," in *11th Conference of the European Chapter of the Association for Computational Linguistics*, 2006.

[16] T. P. Adewumi, F. Liwicki, and M. Liwicki, "The challenge of diacritics in yoruba embeddings," *arXiv preprint arXiv:2011.07605*, 2020.

[17] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Corpora compared: The case of the swedish gigaword & wikipedia corpora," *arXiv preprint arXiv:2011.03281*, 2020.

[18] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," *arXiv preprint arXiv:1712.09405*, 2017.

[19] A. Gatt and E. Krahmer, "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation," *Journal of Artificial Intelligence Research*, vol. 61, pp. 65–170, 2018.

[20] S. Gehrmann, T. Adewumi, K. Aggarwal, P. S. Ammanamanchi, A. Anuoluwapo, A. Bosselut, K. R. Chandu, M. Clinciu, D. Das, K. D. Dhole, *et al.*, "The gem benchmark: Natural language generation, its evaluation and metrics," *arXiv preprint arXiv:2102.01672*, 2021.

[21] B. Chiu, A. Korhonen, and S. Pyysalo, "Intrinsic evaluation of word vectors fails to predict extrinsic performance," in *Proceedings of the 1st workshop on evaluating vector-space representations for NLP*, pp. 1–6, 2016.

[22] B. Wang, A. Wang, F. Chen, Y. Wang, and C.-C. J. Kuo, "Evaluating word embedding models: methods and experimental results," *APSIPA Transactions on Signal and Information Processing*, vol. 8, 2019.

[23] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin, "Placing search in context: The concept revisited," in *Proceedings of the 10th international conference on World Wide Web*, pp. 406–414, 2001.

[24] M. Faruqui, Y. Tsvetkov, P. Rastogi, and C. Dyer, "Problems with evaluation of word embeddings using word similarity tasks," *arXiv preprint arXiv:1605.02276*, 2016.

[25] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Exploring swedish & english fasttext embeddings with the transformer," *arXiv preprint arXiv:2007.16007*, 2020.

[26] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[27] W. Byeon, T. M. Breuel, F. Raue, and M. Liwicki, "Scene labeling with lstm recurrent neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3547–3555, 2015.

[28] F. Simistira, A. Ul-Hassan, V. Papavassiliou, B. Gatos, V. Katsouros, and M. Liwicki, "Recognition of historical greek polytonic scripts using lstm networks," in *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pp. 766–770, IEEE, 2015.

[29] N. Abid, A. ul Hasan, and F. Shafait, "Deepparse: A trainable postal address parser," in *2018 Digital Image Computing: Techniques and Applications (DICTA)*, pp. 1–8, IEEE, 2018.

[30] D. M. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," *arXiv preprint arXiv:2010.16061*, 2020.

[31] T. P. Adewumi, "Inner loop program construct: A faster way for program execution," *Open Computer Science*, vol. 8, no. 1, pp. 115–122, 2018.

[32] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning word vectors for 157 languages," *arXiv preprint arXiv:1802.06893*, 2018.

[33] G. Calmettes, G. B. Drummond, and S. L. Vowler, "Making do with what we have: use your bootstraps," *Advances in physiology education*, vol. 36, no. 3, pp. 177–180, 2012.

[34] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Word2vec: Optimal hyper-parameters and their impact on nlp downstream tasks," *arXiv preprint arXiv:2003.11645*, 2020.

[35] Y. Zhang, S. Sun, M. Galley, Y.-C. Chen, C. Brockett, X. Gao, J. Gao, J. Liu, and B. Dolan, "Dialogpt: Large-scale generative pre-training for conversational response generation," *arXiv preprint arXiv:1911.00536*, 2019.

[36] O. Olabiyi and E. T. Mueller, "Multiturn dialogue response generation with autoregressive transformer models," *arXiv preprint arXiv:1908.01841*, 2019.

Part II

Paper A

Word2Vec: Optimal Hyper-Parameters and Their Impact on NLP Downstream Tasks

Authors:
Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Reformatted version of paper submitted.

Undergoing review

Word2Vec: Optimal Hyper-Parameters and Their Impact on NLP Downstream Tasks

Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Abstract

Word2Vec is a prominent model for natural language processing (NLP) tasks. Similar inspiration is found in distributed embeddings for new state-of-the-art (SotA) deep neural networks. However, wrong combination of hyper-parameters can produce poor quality vectors. The objective of this work is to empirically show optimal combination of hyper-parameters exists and evaluate various combinations. We compare them with the released, pre-trained original word2vec model. Both intrinsic and extrinsic (downstream) evaluations, including named entity recognition (NER) and sentiment analysis (SA) were carried out. The downstream tasks reveal that the best model is usually task-specific, high analogy scores don't necessarily correlate positively with F1 scores and the same applies to the focus on data alone. Increasing vector dimension size after a point leads to poor quality or performance. If ethical considerations to save time, energy and the environment are made, then reasonably smaller corpora may do just as well or even better in some cases. Besides, using a small corpus, we obtain better WordSim scores, corresponding Spearman correlation and better downstream performances (with significance tests) compared to the original model, trained on a 100 billion-word corpus.

1 Introduction

There have been many implementations of the word2vec model in either of the two architectures it provides: continuous skipgram and continuous bag-of-words (CBoW) [1]. Similar distributed models of word or subword embeddings (or vector representations) find usage in SotA, deep neural networks like bidirectional encoder representations from transformers (BERT) and its successors [2, 3, 4]. BERT generates contextual representations of words after been trained for extended periods on large corpora, unsupervised, using the attention mechanisms [5]. Unsupervised learning provides feature representations using large unlabelled corpora [6].

It has been observed that various hyper-parameter combinations have been used in different research involving word2vec, after its release, with the possibility of many of them being sub-optimal [7, 8, 9]. Therefore, the authors seek to address the research question: what is the optimal combination of word2vec hyper-parameters for intrinsic

and extrinsic NLP purposes, specifically NER and SA? There are astronomically high numbers of combinations of hyper-parameters possible for neural networks, even with just a few layers [10]. Hence, the scope of our extensive, empirical work over three English corpora is on dimension size, training epochs, window size and vocabulary size for the training algorithms (hierarchical softmax and negative sampling) of both skipgram and CBoW.

The objective of this work is to determine the optimal combinations of word2vec hyper-parameters for intrinsic evaluation (semantic and syntactic analogies) and a few extrinsic evaluation tasks [11, 12]. It is not our objective in this work to set new SotA results. Some main contributions of this research are the empirical establishment of optimal combinations of word2vec hyper-parameters for NLP tasks, discovering the behaviour of quality of vectors vis-a-vis increasing dimensions and the confirmation of embeddings performance being task-specific for the downstream. The rest of this paper is organised as follows: related work, materials and methods used, experimental that describes experiments performed, results and discussion that present final results, and conclusion.

2 Related Work

Breaking away from the non-distributed (high-dimensional, sparse) representations of words, typical of traditional bag-of-words or one-hot-encoding [13], [1] created word2vec. Word2Vec consists of two shallow neural network architectures: continuous skipgram and CBoW. It uses distributed (low-dimensional, dense) representations of words that group similar words. This new model traded the complexity of deep neural network architectures, by other researchers, for more efficient training over large corpora. Its architectures have two training algorithms: negative sampling and hierarchical softmax [14]. The released model was trained on Google news dataset of 100 billion words. Implementations of the model have been undertaken by researchers in the programming languages Python and C++, though the original was done in C [15]. The Python implementations are slower to train, being an interpreted langauge [16, 17].

Continuous skipgram predicts (by maximizing classification of) words before and after the center word, for a given range. Since distant words are less connected to a center word in a sentence, less weight is assigned to such distant words in training. CBoW, on the other hand, uses words from the history and future in a sequence, with the objective of correctly classifying the target word in the middle. It works by projecting all history or future words within a chosen window into the same position, averaging their vectors. Hence, the order of words in the history or future does not influence the averaged vector. This is similar to the traditional bag-of-words. A log-linear classifier is used in both architectures [1]. In further work, they extended the model to be able to do phrase representations and subsample frequent words [14]. Earlier models like latent dirichlet allocation (LDA) and latent semantic analysis (LSA) exist and effectively achieve low dimensional vectors by matrix factorization [18, 10].

It's been shown that word vectors are beneficial for NLP tasks [13], such as SA and NER. Besides, [1] showed with vector space algebra that relationships among words

can be evaluated, expressing the quality of vectors produced from the model. The famous, semantic example: *vector("King") - vector("Man") + vector("Woman")* ≈ *vector("Queen")* can be verified using cosine distance. Syntactic relationship examples include plural verbs and past tense, among others. WordSimilarity-353 (WordSim) test set is another analysis tool for word vectors [19]. Unlike Google analogy score, which is based on vector space algebra, WordSim is based on human expert-assigned semantic similarity on two sets of English word pairs. Both tools measure embedding quality, with a scaled score of 1 being the highest (very much similar or exact, in Google analogy case).

Like word embeddings, subword representations have proven to be helpful when dealing with out-of-vocabulary (OOV) words and [20] used such embeddings to guide the parsing of OOV words in their work on meaning representation for robots. Despite their success, word embeddings display biases (as one of their shortcomings) seen in the data they are trained on [21]. Intrinsic tests, in the form of word similarity or analogy tests, reveal meaningful relations among words in embeddings, given the relationship among words in context [1, 22]. However, it is inappropriate to assume such intrinsic tests are sufficient in themselves, just as it is inappropriate to assume one particular downstream test is sufficient to generalise the performance of embeddings on all NLP tasks [23, 24, 25].

[1] tried various hyper-parameters with both architectures of their model, ranging from 50 to 1,000 dimensions, 30,000 to 3,000,000 vocabulary sizes, 1 to 3 epochs, among others. In our work, we extended research to 3,000 dimensions and epochs of 5 and 10. Different observations were noticed from the many trials. They observed diminishing returns after a certain point, despite additional dimensions or larger, unstructured training data. However, quality increased when both dimensions and data size were increased together. Although they pointed out that choice of optimal hyper-parameter configurations depends on the NLP problem at hand, they identified the most important factors as architecture, dimension size, subsampling rate, and the window size. In addition, it has been observed that larger datasets improve the quality of word vectors and, potentially, performance on downstream tasks [26, 1] .

3 Materials and methods

3.1 Datasets

The corpora used for word embeddings are the 2019 English Wiki News Abstract by [27] of about 15MB, 2019 English Simple Wiki (SW) Articles by [28] of about 711MB and the Billion Word (BW) of 3.9GB by [29]. The corpus used for sentiment analysis is the internet movie database (IMDb) of movie reviews by [30] while that for NER is the Groningen Meaning Bank (GMB) by [31], containing 47,959 sentence samples. The IMDb dataset used has a total of 25,000 sentences with half being positive sentiments and the other half being negative sentiments. The GMB dataset has 17 labels, with 9 main labels and 2 context tags. Google (semantic and syntactic) analogy test set by [1] and WordSimilarity-353 (with Spearman correlation) by [19] were chosen for intrinsic evaluations.

Table 1: Upstream hyper-parameter choices

Hyper-parameter	Values
Dimension size	300, 1200, 1800, 2400, 3000
Window size (w)	4, 8
Architecture	Skipgram (s1), CBoW (s0)
Algorithm	H. Softmax (h1), N. Sampling (h0)
Epochs	5, 10

3.2 Embeddings

The hyper-parameters tuned in a grid search for the embeddings are given in table 1. The models were generated in a shared cluster running Ubuntu 16 with 32 CPUs of 32x Intel Xeon 4110 at 2.1GHz. Gensim [15] Python library implementation of word2vec was used. This is because of its relative stability, popular support and to minimize the time required in writing and testing a new implementation in Python from scratch. Our models are available for confirmation and source codes are available on github.[1]

3.3 Downstream Architectures

The downstream experiments were run on a Tesla GPU on a shared DGX cluster running Ubuntu 18. Pytorch deep learning framework was used.

A long short term memory network (LSTM) was trained on the GMB dataset for NER. A BiLSTM network was trained on the IMDb dataset for SA. The BiLSTM includes an additional hidden linear layer before the output layer. Hyper-parameter details of the two networks for the downstream tasks are given in table 2. The metrics for extrinsic evaluation include F1, precision, recall and accuracy scores (in the case of SA).

Table 2: Downstream network hyper-parameters

Archi	Epochs	Hidden Dim	LR	Loss
LSTM	40	128	0.01	Cross Entropy
BiLSTM	20	128 * 2	0.0001	BCELoss

4 Experimental

To form the vocabulary for the embeddings, words occurring less than 5 times in the corpora were dropped, stop words removed using the natural language toolkit (NLTK) [32] and additional data pre-processing carried out. Table 1 describes most hyper-parameters explored for each dataset and notations used. In all, 80 runs (of about 160 minutes)

[1] https://github.com/tosingithub/sdesk

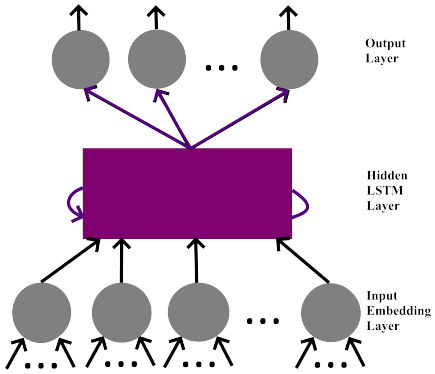

Figure 1: Network architecture for NER

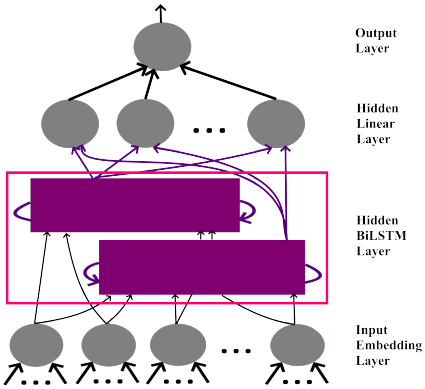

Figure 2: Network architecture for SA

were conducted for the 15MB Wiki Abstract dataset with 80 serialized models totaling 15.136GB while 80 runs (for over 320 hours) were conducted for the 711MB SW dataset, with 80 serialized models totaling over 145GB. Experiments for all combinations for 300 dimensions were conducted on the 3.9GB training set of the BW corpus and additional runs for other dimensions for the window size 8 + skipgram + hierarchical softmax combination to verify the trend of quality of word vectors as dimensions are increased.

Preferably, more than one training instance would have been run per combination for a model and an average taken, however, the long hours involved made this prohibitive. Despite this, we randomly ran a few combinations more than once and confirmed the difference in intrinsic scores were negligible.

For both downstream tasks, the default Pytorch embedding was tested before being replaced by the original (100B) pre-trained embedding and ours. In each case, the dataset was shuffled before training and split in the ratio 70:15:15 for training, dev and test sets. Batch size of 64 was used and Adam as optimizer. For each task, experiments for each embedding was conducted four times and an average value calculated.

5 Results and Discussion

The WordSim result output file from the Gensim Python program always has more than one value reported, including the Spearman correlation. The first value is reported as WordSim score1 in the relevant table. Table 3 summarizes key results from the intrinsic evaluations for 300 dimensions[2]. Table 4 reveals the training time (in hours) and average embedding loading time (in seconds) representative of the various models used. Tables 5 and 6 summarize key results for the extrinsic evaluations. Figures 3, 4, 5, 6 and 7 present line graph of the eight combinations for different dimension sizes for SW, the trend of

[2]The results are to 3 decimal places

Table 3: Scores for 300 dimensions for 10 epochs for SW, BW & 100B corpora.

	w8s1h1	**w8s0h1**	**w8s0h0**	**w8s1h0**	**w4s1h1**	**w4s0h1**	**w4s0h0**	**w4s1h0**
Simple Wiki								
Analogy	0.461	0.269	**0.502**	0.439	0.446	0.243	0.478	0.407
WordSim score1	0.636	0.611	0.654	**0.655**	0.635	0.608	0.620	0.635
Spearman	0.670	0.648	0.667	**0.695**	0.668	0.648	0.629	0.682
Billion Word								
Analogy	0.587	0.376	0.638	0.681	0.556	0.363	0.629	**0.684**
WordSim score1	0.614	0.511	0.599	**0.644**	0.593	0.508	0.597	0.635
Spearman	0.653	0.535	0.618	**0.681**	0.629	0.527	0.615	0.677
Google News - 100B (s1h0)								
Analogy: 0.740			WordSim: 0.624			Spearman: 0.659		

SW and BW corpora over several dimension sizes, analogy score comparison for models across datasets, NER mean F1 scores on the GMB dataset and SA mean F1 scores on the IMDb dataset, respectively. Results for the smallest dataset (Wiki Abstract) are so poor, because of the tiny file size (15MB), there's no reason reporting them here. Hence, we have focused on results from the SW and BW corpora.

Best combination in terms of analogy sometimes changes when corpus size increases, as will be noticed from table 3. In terms of analogy score, for 10 epochs, w8s0h0 performs best while w8s1h0 performs best in terms of WordSim and corresponding Spearman correlation for SW. Meanwhile, increasing the corpus size to BW, w4s1h0 performs best in terms of analogy score while w8s1h0 maintains its position as the best in terms of WordSim and Spearman correlation. Besides considering quality metrics, it can be observed from table 4 that comparative ratio of values between the models is not commensurate with the results in intrinsic or extrinsic values, especially when we consider the amount of time and energy spent, since more training time results in more energy consumption [17].

Table 4: Training & embedding loading time for w8s1h0, w8s1h1 & 100B

Model	**Training (hours)**	**Loading Time (s)**
SW w8s1h0	5.44	1.93
BW w8s1h1	27.22	4.89
GoogleNews (100B)	-	97.73

Information on the length of training time for the original 100B model is not readily available. However, it's interesting to note that it is a skipgram-negative sampling (s1h0) model. Its analogy score, which we tested and report, is confirmed in the original paper [1]. It beats our best models in only analogy score (even for SW), performing worse in others, despite using a much bigger corpus of 3,000,000 vocabulary size and 100 billion

words while SW had vocabulary size of 367,811 and is 711MB. It is very likely our analogy scores will improve when we use a much larger corpus, as can be observed from table 3, which involves just one billion words.

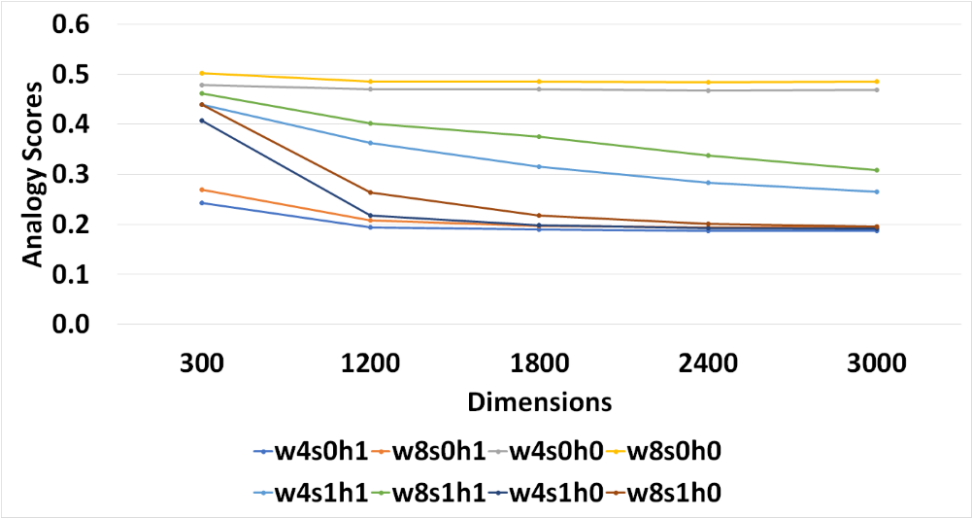

Figure 3: Simple Wiki: Analogy Scores for 10 Epochs (color needed)

With regards to increasing dimension, though the two best combinations in analogy (w8s0h0 & w4s0h0) for SW, as shown in fig. 3, decreased only slightly compared to others, the increased training time and much larger serialized model size render any possible minimal score advantage with higher dimensions undesirable. As can be observed in fig. 4, from 100 dimensions, scores improve but start to drop after over 300 dimensions for SW and after over 400 dimensions for BW, confirming the observation by [1]. This trend is true for all combinations for all tests. Polynomial interpolation may be used to determine the optimal dimension in both corpora.

Table 5: NER Dev and Test sets Mean Results

Metric	**Default**	**100B**	**w8 s0 h0**	**w8 s1 h0**	**BW w4 s1 h0**
	Dev, Test	Dev, Test	Dev, Test	Dev, Test	Dev, Test
F1	0.661, 0.661	**0.679**, 0.676	0.668, 0.669	0.583, 0.676	**0.679, 0.677**
Precision	0.609, 0.608	**0.646, 0.642**	0.636, 0.637	0.553, 0.642	0.644, **0.642**
Recall	0.723, **0.724**	0.716, 0.714	0.704, 0.706	0.618, 0.715	0.717, 0.717

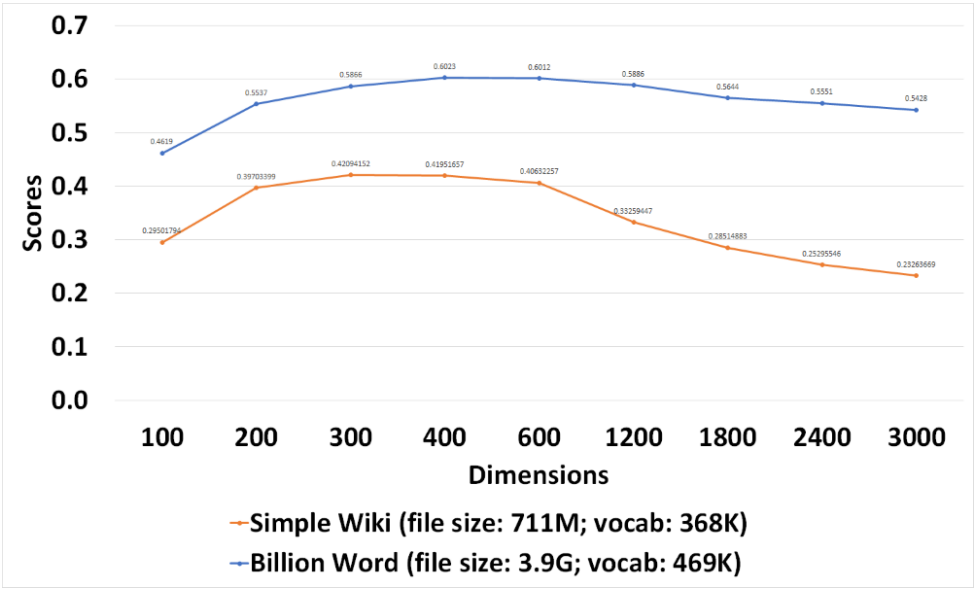

Figure 4: Analogy Scores for w4s1h1 of SW for 5 Epochs & w8s1h1 of BW for 10 epochs (not drawn to scale from 400) (color needed)

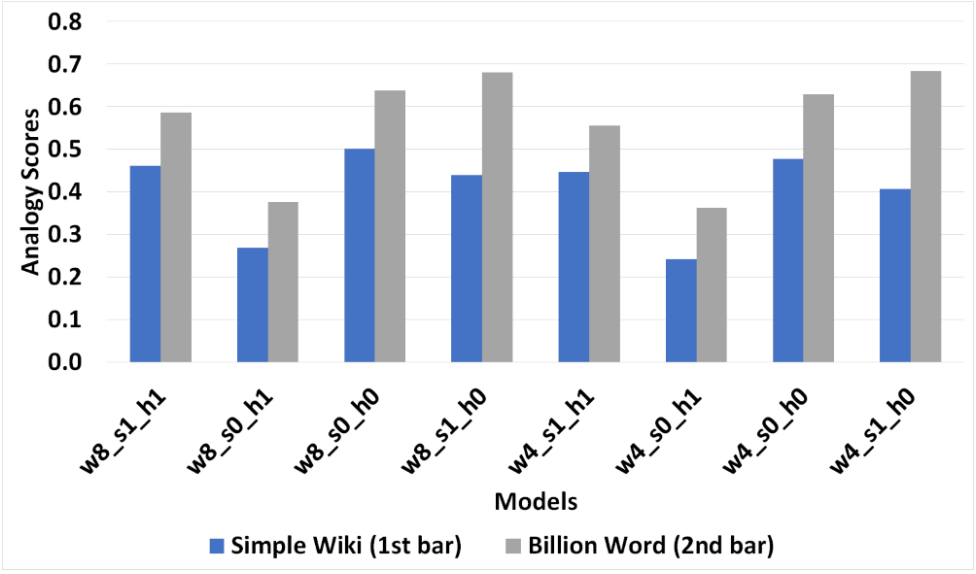

Figure 5: Comparison of 300 dimension models for 10 epochs for SW & BW corpora

Table 6: Sentiment Analysis Dev and Test sets Mean Results

Metric	Default	100B	w8 s0 h0	w8 s1 h0	BW w4 s1 h0
	Dev, Test	Dev, Test	Dev, Test	Dev, Test	Dev, Test
F1	**0.810, 0.805**	0.384, 0.386	0.798, 0.799	0.548, 0.553	0.498, 0.390
Precision	0.805, 0.795	0.6, 0.603	**0.814, 0.811**	0.510, 0.524	0.535, 0.533
Recall	**0.818, 0.816**	0.303, 0.303	0.788, 0.792	0.717, 0.723	0.592, 0.386
Accuracy	**0.807, 0.804**	0.549, 0.55	0.801, 0.802	0.519, 0.522	0.519, 0.517

With regards to NER, most pretrained embeddings outperformed the default Pytorch embedding, with our BW w4s1h0 model (which is best in BW analogy score) performing best in F1 score and closely followed by the 100B model. On the other hand, with regards to SA, Pytorch embedding outperformed the pretrained embeddings but was closely followed by our SW w8s0h0 model (which also had the best SW analogy score). 100B performed second worst of all, despite originating from a very huge corpus. The combinations w8s0h0 & w4s0h0 of SW performed reasonably well in both extrinsic tasks, just as the default Pytorch embedding did.

Significance tests using bootstrap, based on [33], on the results of the differences in means of the 100B & BW w4s1h0 models for NER shows a 95% confidence interval (CI) of [-0.008, 0.01] but [0.274, 0.504] for 100B & SW w8s0h0 for SA. Since one algorithm is involved in the comparisons in each case, unlike multiple algorithms [34], the applied bootstrap approach is adequate. The CI interval for NER includes 0, thus we can conclude the difference was likely due to chance and fail to reject the null hypothesis but the CI for SA does not include 0, thus the difference is unlikely due to chance so we reject the null hypothesis.

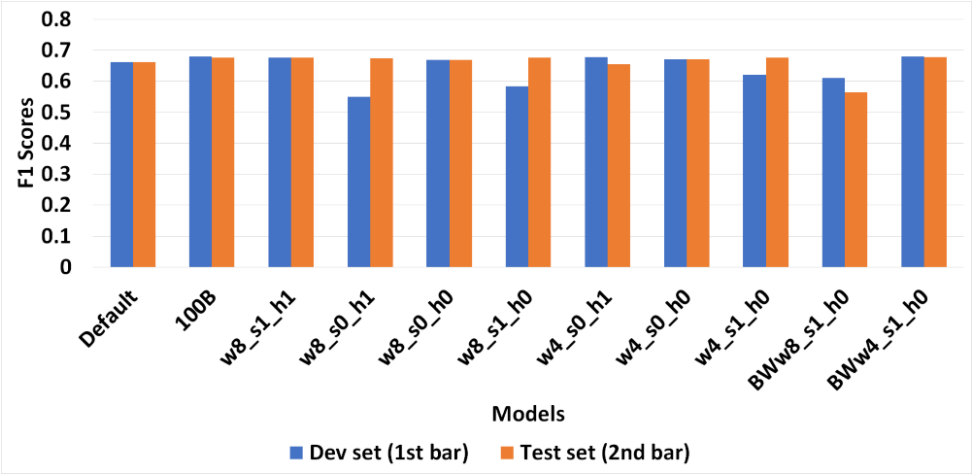

Figure 6: Named Entity Recognition (NER) Mean F1 Scores on GMB Dataset

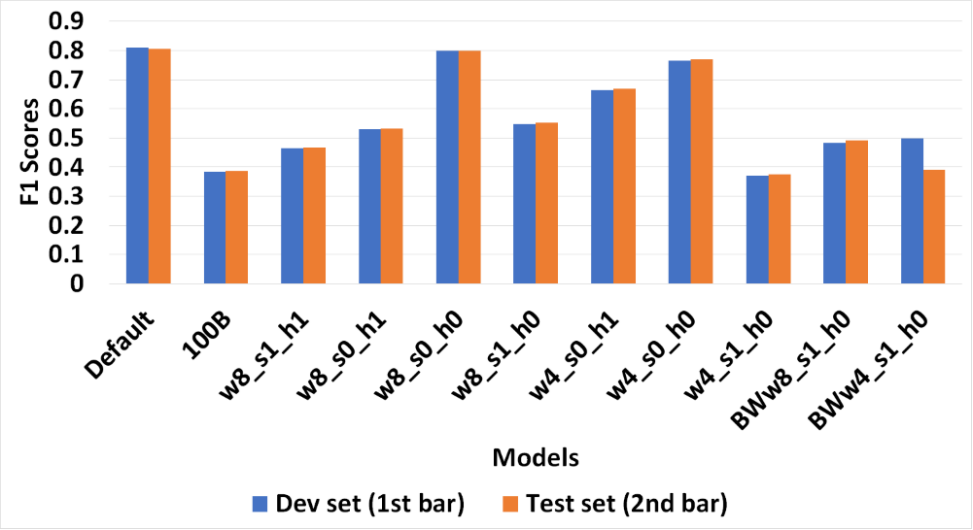

Figure 7: Sentiment Analysis (SA) Mean F1 Scores on IMDb Dataset

6 Conclusions

This work analyses, empirically, optimal combinations of hyper-parameters for embeddings, specifically for word2vec. It further shows that for downstream tasks, like NER and SA, there's no silver bullet! However, some combinations show strong performance across tasks. Performance of embeddings is task-specific and high analogy scores do not necessarily correlate positively with performance on downstream tasks. This point on correlation is somewhat similar to results by [35] and [12]. It was discovered that increasing embedding dimension size depreciates performance after a point. If strong considerations of saving time, energy and the environment are made, then reasonably smaller corpora may suffice or even be better in some cases. The on-going drive by many researchers to use ever-growing data to train deep neural networks can benefit from the findings of this work. Indeed, hyper-parameter choices are very important in neural network systems [10].

Future work that may be investigated are the performance of other architectures of word or sub-word embeddings in SotA networks like BERT (based on a matrix of hyper-parameters), the performance and comparison of embeddings applied to other less-explored languages, and how these embeddings perform in other downstream tasks.

Funding

This work was supported partially by Vinnova under the project number 2019-02996 'Språkmodeller för svenska myndigheter'. They, however, had no involvement in any stage of this work, including study design, interpretation of data and report writing.

References

[1] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[3] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[4] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *arXiv preprint arXiv:1910.10683*, 2019.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[6] M. Längkvist, L. Karlsson, and A. Loutfi, "A review of unsupervised feature learning and deep learning for time-series modeling," *Pattern Recognition Letters*, vol. 42, pp. 11–24, 2014.

[7] B. Dhingra, H. Liu, R. Salakhutdinov, and W. W. Cohen, "A comparative study of word embeddings for reading comprehension," *arXiv preprint arXiv:1703.00993*, 2017.

[8] M. Naili, A. H. Chaibi, and H. H. B. Ghezala, "Comparative study of word embedding methods in topic segmentation," *Procedia computer science*, vol. 112, pp. 340–349, 2017.

[9] Y. Wang, S. Liu, N. Afzal, M. Rastegar-Mojarad, L. Wang, F. Shen, P. Kingsbury, and H. Liu, "A comparison of word embeddings for the biomedical natural language processing," *Journal of biomedical informatics*, vol. 87, pp. 12–20, 2018.

[10] O. Levy, Y. Goldberg, and I. Dagan, "Improving distributional similarity with lessons learned from word embeddings," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 211–225, 2015.

[11] Y. Zhang, Q. Chen, Z. Yang, H. Lin, and Z. Lu, "Biowordvec, improving biomedical word embeddings with subword information and mesh," *Scientific data*, vol. 6, no. 1, pp. 1–9, 2019.

[12] B. Wang, A. Wang, F. Chen, Y. Wang, and C.-C. J. Kuo, "Evaluating word embedding models: methods and experimental results," *APSIPA Transactions on Signal and Information Processing*, vol. 8, 2019.

[13] J. Turian, L. Ratinov, and Y. Bengio, "Word representations: a simple and general method for semi-supervised learning," in *Proceedings of the 48th annual meeting of the association for computational linguistics*, pp. 384–394, Association for Computational Linguistics, 2010.

[14] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, pp. 3111–3119, 2013.

[15] R. Řehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, (Valletta, Malta), pp. 45–50, ELRA, May 2010.

[16] T. P. Adewumi, "Inner loop program construct: A faster way for program execution," *Open Computer Science*, vol. 8, no. 1, pp. 115–122, 2018.

[17] T. P. Adewumi and M. Liwicki, "Inner for-loop for speeding up blockchain mining," *Open Computer Science*, 2019.

[18] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for information science*, vol. 41, no. 6, pp. 391–407, 1990.

[19] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin, "Placing search in context: The concept revisited," *ACM Transactions on information systems*, vol. 20, no. 1, pp. 116–131, 2002.

[20] J. Thomason, A. Padmakumar, J. Sinapov, N. Walker, Y. Jiang, H. Yedidsion, J. Hart, P. Stone, and R. Mooney, "Jointly improving parsing and perception for natural language commands through human-robot dialog," *Journal of Artificial Intelligence Research*, vol. 67, pp. 327–374, 2020.

[21] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai, "Man is to computer programmer as woman is to homemaker? debiasing word embeddings," in *Advances in neural information processing systems*, pp. 4349–4357, 2016.

[22] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[23] A. Gatt and E. Krahmer, "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation," *Journal of Artificial Intelligence Research*, vol. 61, pp. 65–170, 2018.

[24] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Corpora compared: The case of the swedish gigaword & wikipedia corpora," *arXiv preprint arXiv:2011.03281*, 2020.

[25] T. P. Adewumi, F. Liwicki, and M. Liwicki, "The challenge of diacritics in yoruba embeddings," *arXiv preprint arXiv:2011.07605*, 2020.

[26] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Conversational systems in machine learning from the point of view of the philosophy of science—using alime chat and related studies," *Philosophies*, vol. 4, no. 3, p. 41, 2019.

[27] Wikipedia, "Wiki news abstract," 2019.

[28] Wikipedia, "Simple wiki articles," 2019.

[29] C. Chelba, T. Mikolov, M. Schuster, Q. Ge, T. Brants, P. Koehn, and T. Robinson, "One billion word benchmark for measuring progress in statistical language modeling," tech. rep., Google, 2013.

[30] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1*, pp. 142–150, Association for Computational Linguistics, 2011.

[31] J. Bos, V. Basile, K. Evang, N. J. Venhuizen, and J. Bjerva, "The groningen meaning bank," in *Handbook of linguistic annotation*, pp. 463–496, Springer, 2017.

[32] E. Loper and S. Bird, "Nltk: the natural language toolkit," *arXiv preprint cs/0205028*, 2002.

[33] G. Calmettes, G. B. Drummond, and S. L. Vowler, "Making do with what we have: use your bootstraps," *Advances in physiology education*, vol. 36, no. 3, pp. 177–180, 2012.

[34] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine learning research*, vol. 7, no. Jan, pp. 1–30, 2006.

[35] B. Chiu, A. Korhonen, and S. Pyysalo, "Intrinsic evaluation of word vectors fails to predict extrinsic performance," in *Proceedings of the 1st workshop on evaluating vector-space representations for NLP*, pp. 1–6, 2016.

Paper B

Exploring Swedish & English fastText Embeddings for NER with the Transformer

Authors:
Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Reformatted version of paper submitted.

Undergoing review

Exploring Swedish & English fastText Embeddings for NER with the Transformer

Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Abstract

In this paper, our main contributions are that embeddings from relatively smaller corpora can outperform ones from larger corpora and we make the new Swedish analogy test set publicly available. To achieve a good network performance in natural language processing (NLP) downstream tasks, several factors play important roles: dataset size, the right hyper-parameters, and well-trained embeddings. We show that, with the right set of hyper-parameters, good network performance can be reached even on smaller datasets. We evaluate the embeddings at both the intrinsic and extrinsic levels. The embeddings are deployed with the Transformer in named entity recognition (NER) task and significance tests conducted. This is done for both Swedish and English. We obtain better performance in both languages on the downstream task with smaller training data, compared to recently released, Common Crawl versions; and character n-grams appear useful for Swedish, a morphologically rich language.

1 Introduction

The embedding layer of neural networks may be initialized randomly or replaced with pre-trained vectors, which act as lookup tables. One of such pre-trained vector tools include fastText, introduced by Joulin et al. [1]. The main advantages of fastText are speed and competitive performance to state-of-the-art (SotA). Using pre-trained embeddings in deep networks like the Transformer can improve performance. Vaswani et al. (2017) introduced the Transformer, a SotA architecture based on self-attention mechanisms only, and it demonstrated better performance while requiring less time to train [2]. Usually, downstream tasks are applied after pre-training language models on such deep networks [3, 4].

Despite the plethora of embeddings in many languages, there's a dearth of analogy test sets to evaluate many of them, including for Swedish [5, 6, 7, 8]. This is because creating labelled or structured datasets can be expensive in terms of time and attention required. Grave et al. (2018) created 157 different language embeddings but provided analogy test set for only 3 languages: French, Hindi and Polish [9]. An analogy test set, introduced by Mikolov et al. (2013), provides some inclination as to the quality and

likely performance of word embeddings in NLP downstream tasks, such as NER, which is used in this work [10]. The evaluation involves prediction of the second value of a pair of two similar words.

Therefore, key contributions of this work (from its objective) are (i) the new Swedish analogy test set publicly made available[1] for the NLP research community, (ii) optimal English and Swedish embeddings, and (iii) insight into their performance in the NER downstream task. The quality of the Swedish model by Grave et al. (2018) is evaluated, in a first. The embedding hyper-parameters are based on previous research, which used grid search to determine optimal hyper-parameters [11]. The rest of this paper is organised as follows: a brief survey of related work, the methodology used, results and discussion, and the conclusion.

2 Related Work

Distributed representation of words has been around for some time [12]. fastText, based on the original distributed representation by Mikolov et al. (2013), contains two architectures [10]. Its continuous bag of words (CBoW) averages word vectors into text representation, fed into a linear classifier, while the skipgram uses bag of character n-grams for represented words by summing them [13, 1]. The use of subword representations has proven to be helpful when dealing with out-of-vocabulary (OOV) words. Indeed, Thomason et al. (2020) used word embeddings to guide the parsing of OOV words in their work on meaning representation for robots [14] .

Despite the potential advantage of subword vectors, Bojanowski et al (2017) observed that using character n-gram was less useful for English compared to some other languages they had explored after a few of the languages were evaluated using different datasets [13]. It is doubtful if comparison of their results obtained across languages is truly justified, given that different Wikipedia corpora, possibly of different sizes, were trained and tested on different analogy datasets. This risk was observed by other researchers while working with English and German embeddings, for which they took measures [15].

WordSimilarity-353 (WordSim) test set is another analysis tool for word vectors [16]. It is based on human expert-assigned semantic similarity on two sets of English word pairs. This is unlike analogy score, based on vector space algebra. Both are used to measure intrinsic embedding quality. Despite their weaknesses, they have been shown to reveal somewhat meaningful relationships among words in embeddings [10, 17]. It is misleading to assume such intrinsic tests are sufficient in themselves, just as it is misleading to assume one particular extrinsic test is sufficient to generalise the performance of embeddings on all NLP tasks [18, 19, 11]. For Swedish, a common evaluation resource for words is SALDO [20], which is a lexical-semantic resource that links words by their associations. SALDO extends SAL (Svenskt associationslexikon, a set of classified synonyms) with inflectional morphological information [20, 21]. QVEC-CCA may be used as an intrinsic evaluation metric with features from language resource like SALDO [22, 6].

[1] github.com/tosingithub/tdesk

Joulin et al. (2016) noted that other implementations of their fastText model could be much slower [1]. Indeed, implementations in Python, an interpreted language, are expected to be slower and will use up more energy resources, compared to the original C++ implementation [23, 24]. The English and Swedish language models by Grave et al. (2018) were trained on Common Crawl & Wikipedia datasets, using CBoW of 300 dimensions, with character n-grams of length 5 and window size 5 [9]. These are the embeddings we compare with in this work. Common Crawl contains petabytes of data, resulting in 630 billion words after preprocessing in a previous use [25].

The Transformer, in its original form, maintains an encoder-decoder architecture [2]. An input sequence is mapped to a sequence of continuous representations by the encoder. Then, the decoder makes auto-regressive output sequence of symbols, one at a time, utilizing the previously generated symbols as extra input for the next. Self-attention, in neural networks, computes a representation of various positions of a sequence and this is what the Transformer architecture employs [2]. The Transformer architecture, in one form or the other, has been utilized in recent SotA results [4, 3].

3 Methodology

3.1 Upstream

All pre-trained models in English and Swedish were generated using the original C++ implementation [9]. This forestalls using any sub-optimal, third-party implementations. They were run on a shared DGX cluster running Ubuntu 18 with 80 CPUs. Gensim Python library program was used to evaluate all models against their corresponding analogy test sets. Some of the default hyper-parameter settings were retained [13]. All models are 300 dimensions and trained for 10 epochs. The lower and upper boundaries for the character n-gram were 3 and 6, respectivley. Table 1 identifies other hyper-parameters (and notations used in subsequent tables).

Both the English and Swedish training datasets used are 2019 Wikipedia dumps of 27G (4.86B words) and 4G (767M words), respectively, after pre-processing [26, 27]. They were pre-processed using the recommended script [9]. It would have been ideal to run each training multiple times to obtain averages but because of the limited time involved, a work-around was adopted, which was to run a few random models twice to ascertain if there were major differences per model. It was established that differences were little enough to accept a single run per model. Besides, each run took hours within the range of about 2 and 36 hours and there were 32 pre-trained models to be generated: 8 English subword and no-subword (word2vec) models each and 8 Swedish subword and no-subword models each.

3.2 Downstream

The downstream tasks were run on the same cluster mentioned earlier but on Tesla V100 GPU. The models and source codes are available[1]. Selected pre-trained embed-

Hyper-parameter	Values
Window size (w)	4, 8
Architecture	Skipgram (s1), CBoW (s0)
Loss Function	H. Softmax (h1), N. Sampling (h0)

Table 1: Hyper-parameter choices

dings were evaluated, for both languages, using the Transformer Encoder architecture in PyTorch. This is without language model pre-training of the Transformer. There are other models/architectures that can be applied to NER, such as conditional random field (CRF)-based models [28, 29] but this can be left to future work. Two corpora were used for the NER downstream task: Groningen Meaning Bank (GMB) for the Englsih NER [30] and the Stockholm Internet Corpus (SiC) [31]. GMB contains 47,959 sentence samples, with 17 labels from 9 main labels and 2 context tags. SiC contains 13,562 samples and follows the CoNLL & SUC 3.0 (Stockholm-Umeå Corpus) formats. It has 3 main tags and 8 types, resulting in 17 possible label combinations, however, in practice, 14 labels are currently represented in the corpus.

In both language cases of the NER experiments, the default PyTorch embedding was tested before being replaced by the pre-trained embeddings, with frozen weights. In each case, the dataset was shuffled before training and split in the ratio 70:15:15 for training, dev and test sets. Three hyper-parameters were tuned using SigOpt (Bayesian hyper-parameter optimization tool) for 45 combinations (or observation budget) over the network optimizer (between Adam & RMSProp), Transformer layers (6-12) and attention heads (2-6) [32]. This approach eliminates the need to explore all possible combinations in a grid search. For the English NER, SigOpt optimized and reported the following values: 7 layers, 3 heads and Adam optimizer. These values were then kept constant for all other embeddings in English. The same was done for the Swedish NER after optimized values obtained were 8 layers, 2 heads and RMSProp optimizer. Batch size of 64 was used and each experiment conducted five times and average values reported. Each run of experiment was for 20 epochs. However, after validation at each epoch, the model is saved, if it has lower loss than a previous value, thereby avoiding overfitting. The saved model is then used to evaluate the test set.

3.3 Swedish analogy test set

The Swedish analogy test set follows the format of the original Google version. The original has been observed to be slightly unbalanced, having 8,869 semantic samples and 10,675 syntactic samples (making a total of 19,544). The Swedish set is bigger and balanced across the 2 major categories, having a total of 20,637, made up of 10,380 semantic and 10,257 syntactic samples. It is also roughly balanced across the syntactic subsections but the *capital-world* has the largest proportion of samples in the semantic subsection. This is because of the difficulty involved in obtaining world currencies in Swedish and the limited nomenclature of family members. A similar difficulty was experienced by Venekoski & Vankka (2017), who noted that not all words in the original

Google analogy test set can be directly translated to other languages, while creating a much smaller Finnish version. In all, there are 5 semantic subsections and 6 syntactic subsections. Table 2 presents further details on the test set. It was constructed, partly using the samples in the English version, with the help of tools dedicated to Swedish dictionary/translation[2] and was proof-read for corrections by two native speakers (with a percentage agreement of 98.93%). New, relevant entries were also added. The famous sample in the family subsection of the semantic section is: *kung drottning man kvinna*.

Semantic	**Syntactic**
capital-common-countries (342)	gram2-opposite (2,652)
capital-world (7,832)	gram3-comparative (2,162)
currency (42)	gram4-superlative (1,980)
city-in-state (1,892)	gram6-nationality-adjective (12)
family (272)	gram7-past-tense (1,891)
	gram8-plural (1,560)

Table 2: Swedish analogy test set details

4 Results & Discussion

The WordSim result output file from the Gensim Python program always has more than one value reported, including the Spearman correlation. The first value is reported as WordSim score1 in the relevant table. Intrinsic results for the pre-trained models are given in table 3. An important trend that can be observed is the higher scores for skipgram-negative sampling in all the cases (English & Swedish), except one. This appears to confirm previous research [10, 11]. It is noteworthy that the released, original pre-trained word2vec model was of the same combination [10]. This English word2vec (no-subword) embedding was trained on GoogleNews dataset of 100 billion words and represented as *'GN'* in the table [10]. The English subword embeddings have 5 models with higher analogy scores than their word2vec equivalent, out of 8. The WordSim and corresponding Spearman correlation for English word2vec models were higher than their corresponding subword models in all cases, except one. It may not be proper to compare the scores of the English to the Swedish models since both were based on different test sets of varying sizes.

Given the observation that using character n-gram was less useful for English than some other languages, it's not expected that the scores will follow a similar trend for all languages [13]. In addition, accuracy falls for morphologically complex languages, like German, making analogy predictions difficult [15]. While working on Finnish embeddings, it was observed that fastText (subword) CBoW had lower analogy score than

[2]https://bab.la & https://en.wiktionary.org/wiki/

word2vec CBoW while fastText skipgram had higher score than word2vec skipgram, even for zero OOV words [8].

Indeed, determining the best pre-trained model in each category requires the additional step of applying them to downstream tasks, in this case NER [33]. Tables 4 & 5 present the results of the NER task for the selected English & Swedish embeddings, respectively. The embeddings by Grave et al. (2018), trained on the larger Common Crawl & Wikipedia, are represented by '*Gr*' in the tables. It can be observed that for English, the word2vec w8s0h0 embedding outperformed the subword embedding: *Gr*. The Swedish subword embedding, *Gr*, is also outperformed by the subword embeddings the authors created. Importantly, the subword versions outperform the word2vec ones, implying the character n-grams may be useful for Swedish. In both language cases, the good performance of PyTorch default embedding is noticeable.

	Skipgram (s1)				**CBoW (s0)**					
	H. S. (h1)		**N. S. (h0)**		**H. S. (h1)**		**N. S. (h0)**		**Gr**	**GN**
window (w)	**4**	**8**	**4**	**8**	**4**	**8**	**4**	**8**		
Subword %										
Analogy	62.6	58.8	74.4	69.8	67.2	68.7	71.6	71	**82.6**	
WordSim score1	64.8	66.3	69.9	**70**	62.6	66.2	47.3	51.1	68.5	
Spearman	67.6	69.4	**74.3**	73.6	65.3	70.3	45.3	49.5	70.2	
Word2Vec %										
Analogy	61.3	58.3	73.5	70.4	59.7	61.9	**76.2**	75.4		74
WordSim score1	66.3	67.3	69.6	**70.1**	64.1	66.7	65.4	67.5		62.4
Spearman	70	70.9	74.5	**74.7**	68.2	71.2	66.9	69.4		65.9
Swedish										
Subword %	45.05	39.99	53.53	53.36	26.5	23.93	36.79	35.89	**60.9**	
Word2Vec %	45.53	41.21	**58.25**	57.30	28.02	28.04	52.81	55.64		

Table 3: Intrinsic Scores - English & Swedish (highest score/row in bold)

Significance tests, using bootstrap [34], on the results of the differences in means of the English *Gr* & word2vec w8s0h0 models, show a 95% confidence interval (CI) of [0.0003, 0.1674] but [-0.3257, 0.169] for Swedish *Gr* & subword w4s1h1. The CI interval for English does not include 0, though the lower limit is small, thus we can conclude the difference is unlikely due to chance but the CI for Swedish includes 0, thus the difference is likely due to chance.

4.1 Embedding Qualitative Assessment

Qualitative assessment of the Swedish model (subword w4s1h1) in one instance is given in table 6, for randomly selected input.

					Word2Vec (W)						Subword			
Metric	**Default**		**Gr**		**w8s0h0**		**w4s0h0**		**w4s1h0**		**w4s0h0**		**w8s1h1**	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test
F1	0.719	**0.723**	0.588	0.6602	0.719	0.720	0.715	0.716	0.714	0.716	0.695	0.668	0.592	0.684
Precision	0.685	0.69	0.564	0.634	0.689	**0.691**	0.686	0.688	0.684	0.686	0.664	0.64	0.567	0.656
Recall	0.756	**0.759**	0.615	0.689	0.751	0.752	0.747	0.747	0.748	0.748	0.729	0.7	0.62	0.713

Table 4: English NER Mean Scores

					Word2Vec (W)				Subword					
Metric	**Default**		**Gr**		**w4s1h0**		**w8s0h1**		**w4s1h1**		**w4s1h0**		**w8s0h1**	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test
F1	0.487	**0.675**	0.441	0.568	0.574	0.344	0.477	0.429	0.507	0.649	0.492	0.591	0.486	0.623
Precision	0.51	0.745	0.682	**0.856**	0.704	0.549	0.626	0.669	0.647	0.821	0.658	0.752	0.626	0.802
Recall	0.471	**0.633**	0.331	0.44	0.489	0.265	0.398	0.325	0.420	0.543	0.398	0.5	0.402	0.524

Table 5: Swedish NER Mean Scores

4.2 Learning Qualitative Assessment

It was observed that learning occurs faster with the Transformer than the LSTM, which was used in an earlier work. Tables 7 & 8 provide examples for both languages. In one instance, in the English case, learning almost correctly occurs by epoch 5. We observed that most times it's earlier. A similar occurrence is observed with Swedish. The learning is not always 100% correct, though.

5 Conclusion

This work has presented optimal fastText embeddings in Swedish and English for NLP purposes. It has also presented the first Swedish analogy test set for intrinsic evaluation of Swedish embeddings. The intrinsic evaluation shows the trend of better performance with skipgram-negative sampling pre-trained models across the two languages. We also observe that for downstream evaluation for English, the word2vec embedding: CBoW-negative sampling of window size 8, like its other counterparts, outperform the subword embedding of the bigger Common Crawl dataset. From the results, it may be that WordSim makes better predictions of the performance on downstream tasks. The Swedish subword embeddings outperform the word2vec versions, implying that character n-grams may be useful for Swedish, a morphologically rich language. Also, they outperform the subword embedding of the larger Common Crawl dataset.

Merely increasing training dataset size does not equate to better performance and optimal hyper-parameters can improve performance [11]. Future work can evaluate embeddings of language model pre-training of the Transformer-based SotA models and other downstream tasks. Other Machine Learning frameworks may also be evaluated.

Figure 1: English NER mean F1 scores

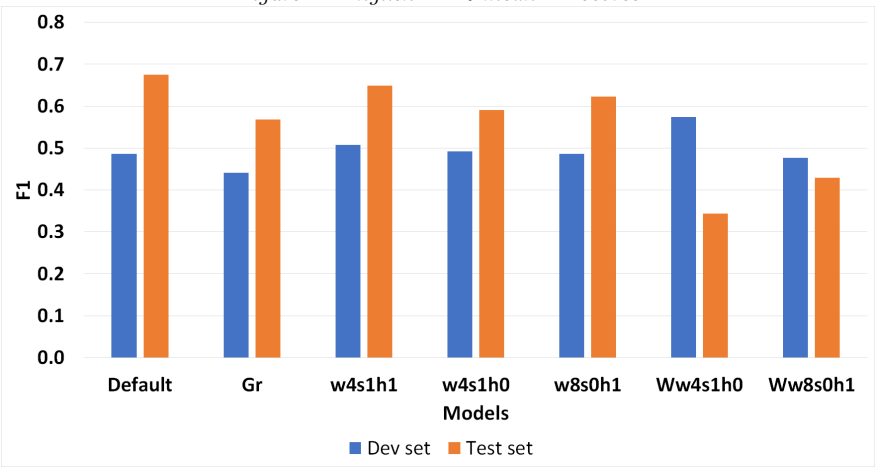

Figure 2: Swedish NER mean F1 scores

Nearest Neighbor/ Analogy Query	Result
syster	halvsyster (0.8688), systerdotter (0.8599), ...
rom - italien + kairo	egypten (0.4889), norditalien (0.4317), ...

Table 6: Qualitative assessment of Swedish w4s1h1 model

Acknowledgment

The authors wish to thank Carl Borngrund and Karl Ekström for their very useful help in proof-reading the analogy set. The work in this project is partially funded by Vinnova under the project number 2019-02996 "Språkmodeller för svenska myndigheter".

	Sample Sentence Tokens/ Tags													
Sentence:	Turkey	's	Foreign	Ministry	says	several	of	its	nationals	were	killed	Friday	in	an
	ambush	in	the	northern	Iraqi	city	of	Mosul	.					
True Tags	B-org	I-org	I-org	I-org	O	O	O	O	O	O	O	B-tim	O	O
	O	O	O	O	B-gpe	O	O	B-geo	O					
Tags@Epoch 1	B-geo	O	O	O	O	O	O	O	O	O	O	B-tim	O	O
	O	O	O	O	B-gpe	O	O	B-geo	O					
Tags@Epoch 2	B-geo	O	O	I-org	O	O	O	O	O	O	O	B-tim	O	O
	O	O	O	O	B-gpe	O	O	B-geo	O					
Tags@Epoch 5	B-org	O	I-org	I-org	O	O	O	O	O	O	O	B-tim	O	O
	O	O	O	O	B-gpe	O	O	B-geo	O					

Table 7: English Learning Sample

	Sample Sentence Tokens/ Tags									
Sentence:	Även	kollat	upp	lite	tågresor	till	Borlänge	i	sommar	!
True Tags	O	O	O	O	O	O	Bplace	O	O	O
Tags@Epoch 1	O	O	O	O	O	O	O	O	O	O
Tags@Epoch 2	O	O	O	O	O	O	Bplace	O	O	O

Table 8: Swedish Learning Sample

References

[1] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," *arXiv preprint arXiv:1607.01759*, 2016.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[3] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners," *arXiv preprint arXiv:2005.14165*, 2020.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[5] R. Al-Rfou, B. Perozzi, and S. Skiena, "Polyglot: Distributed word representations for multilingual nlp," *arXiv preprint arXiv:1307.1662*, 2013.

[6] P. Fallgren, J. Segeblad, and M. Kuhlmann, "Towards a standard dataset of swedish word vectors," in *Sixth Swedish Language Technology Conference (SLTC), Umeå 17-18 nov 2016*, 2016.

[7] R. Précenth, "Word embeddings and gender stereotypes in swedish and english," 2019.

[8] V. Venekoski and J. Vankka, "Finnish resources for evaluating language model semantics," in *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pp. 231–236, 2017.

[9] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning word vectors for 157 languages," *arXiv preprint arXiv:1802.06893*, 2018.

[10] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[11] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Word2vec: Optimal hyper-parameters and their impact on nlp downstream tasks," *arXiv preprint arXiv:2003.11645*, 2020.

[12] G. E. Hinton *et al.*, "Learning distributed representations of concepts," in *Proceedings of the eighth annual conference of the cognitive science society*, vol. 1, p. 12, Amherst, MA, 1986.

[13] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[14] J. Thomason, A. Padmakumar, J. Sinapov, N. Walker, Y. Jiang, H. Yedidsion, J. Hart, P. Stone, and R. Mooney, "Jointly improving parsing and perception for natural language commands through human-robot dialog," *Journal of Artificial Intelligence Research*, vol. 67, pp. 327–374, 2020.

[15] M. Köper, C. Scheible, and S. S. im Walde, "Multilingual reliability and "semantic" structure of continuous word spaces," in *Proceedings of the 11th international conference on computational semantics*, pp. 40–45, 2015.

[16] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin, "Placing search in context: The concept revisited," *ACM Transactions on information systems*, vol. 20, no. 1, pp. 116–131, 2002.

[17] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[18] A. Gatt and E. Krahmer, "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation," *Journal of Artificial Intelligence Research*, vol. 61, pp. 65–170, 2018.

[19] M. Faruqui, Y. Tsvetkov, P. Rastogi, and C. Dyer, "Problems with evaluation of word embeddings using word similarity tasks," *arXiv preprint arXiv:1605.02276*, 2016.

[20] L. Borin, M. Forsberg, and L. Lönngren, "Saldo: a touch of yin to wordnet's yang," *Language resources and evaluation*, vol. 47, no. 4, pp. 1191–1211, 2013.

[21] S. R. Eide, N. Tahmasebi, and L. Borin, "The swedish culturomics gigaword corpus: A one billion word swedish reference dataset for nlp," in *Digital Humanities 2016. From Digitization to Knowledge 2016: Resources and Methods for Semantic Processing of Digital Works/Texts, Proceedings of the Workshop, July 11, 2016, Krakow, Poland*, no. 126, pp. 8–12, Linköping University Electronic Press, 2016.

[22] Y. Tsvetkov, M. Faruqui, and C. Dyer, "Correlation-based intrinsic evaluation of word vector representations," *arXiv preprint arXiv:1606.06710*, 2016.

[23] T. P. Adewumi, "Inner loop program construct: A faster way for program execution," *Open Computer Science*, vol. 8, no. 1, pp. 115–122, 2018.

[24] T. P. Adewumi and M. Liwicki, "Inner for-loop for speeding up blockchain mining," *Open Computer Science*, vol. 10, no. 1, pp. 42–47, 2020.

[25] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," *arXiv preprint arXiv:1712.09405*, 2017.

[26] Wikipedia, "English wikipedia multistream articles," 2019.

[27] Wikipedia, "Swedish wikipedia multistream articles," 2019.

[28] J. R. Finkel, T. Grenager, and C. D. Manning, "Incorporating non-local information into information extraction systems by gibbs sampling," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pp. 363–370, 2005.

[29] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit," in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pp. 55–60, 2014.

[30] J. Bos, V. Basile, K. Evang, N. J. Venhuizen, and J. Bjerva, "The groningen meaning bank," in *Handbook of linguistic annotation*, pp. 463–496, Springer, 2017.

[31] U. Stockholm, "Stockholm internet corpus," 2017.

[32] R. Martinez-Cantin, K. Tee, and M. McCourt, "Practical bayesian optimization in the presence of outliers," in *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics* (A. Storkey and F. Perez-Cruz, eds.), vol. 84 of *Proceedings of Machine Learning Research*, (Playa Blanca, Lanzarote, Canary Islands), p. 1722–1731, PMLR, 09–11 Apr 2018.

[33] B. Chiu, A. Korhonen, and S. Pyysalo, "Intrinsic evaluation of word vectors fails to predict extrinsic performance," in *Proceedings of the 1st workshop on evaluating vector-space representations for NLP*, pp. 1–6, 2016.

[34] G. Calmettes, G. B. Drummond, and S. L. Vowler, "Making do with what we have: use your bootstraps," *Advances in physiology education*, vol. 36, no. 3, pp. 177–180, 2012.

Paper C

Corpora Compared: The Case of the Swedish Gigaword & Wikipedia Corpora

Authors:
Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Reformatted version of paper accepted at:
Swedish Language Technology Conference (SLTC) 2020

© 2020, The Publisher, Reprinted with permission.

Corpora Compared: The Case of the Swedish Gigaword & Wikipedia Corpora

Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Abstract

In this work, we show that the difference in performance of embeddings from differently sourced data for a given language can be due to other factors besides data size. Natural language processing (NLP) tasks usually perform better with embeddings from bigger corpora. However, broadness of covered domain and noise can play important roles. We evaluate embeddings based on two Swedish corpora: The Gigaword and Wikipedia, in analogy (intrinsic) tests and discover that the embeddings from the Wikipedia corpus generally outperform those from the Gigaword corpus, which is a bigger corpus. Downstream tests will be required to have a definite evaluation.

1 Introduction

It is generally observed that more data bring about better performance in Machine Learning (ML) tasks [1, 2]. What may not be very clear is the behaviour of variance of homogeneity in datasets. It is always better to have a balanced or broad-based dataset or avoid an overly-represented topic within a dataset [2]. Furthermore, noise (or contamination) in data can reduce performance [3]. However, not all noise is bad. Indeed, noise may be helpful [2].

In this work, we compare embeddings (in analogy test) from two Swedish corpora: The Gigaword and Wikipedia. The Gigaword corpus by [4] contains data from different genre, covering about 7 decades since the 1950s. Meanwhile the Wikipedia is a collection of articles on many, various subjects [5].

Word similarity or analogy tests, despite their weaknesses, have been shown to reveal somewhat meaningful relationships among words in embeddings, given the relationship among words in context [6, 7]. It is misleading to assume such intrinsic tests are sufficient in themselves, just as it is misleading to assume one particular extrinsic (downstream) test is sufficient to generalise the performance of embeddings on all NLP tasks [8, 9, 10].

The research question being addressed in this work is: does bigger corpus size automatically mean better performance for differently-sourced Swedish corpora? The contribution this work brings is the insight into the differences in the performance of the Swedish embeddings of the Gigaword and Wikipedia corpora, despite the over 40% additional size of the Gigaword corpus. Furthermore, this work will, possibly, enable researchers seek out ways to improve the Gigaword corpus, and indeed similar corpora, if NLP downstream tasks confirm the relative better performance of embeddings from the

Wikipedia corpus. The following sections include related work, methodology, results & discussion and conclusion.

2 Related Work

[4] created the Swedish corpus with at least one billion words. It covers fiction, government, news, science and social media from the 1950s. The sentences of the first six lines of the content of this Gigaword corpus are:

> 1 knippa dill
> patrik andersson
> TV : Danska Sidse Babett Knudsen har prisats på tv-festivalen i Monte Carlo för rollen
> i dramaserien Borgen .
> Hon sköts med ett skott i huvudet , men tog sig fram till porten och ringde på .
> I början av juni tog hon examen från den tvååriga YH-utbildning , som hon flyttade upp till huvudstaden för att gå .
> Det blev kaos , folk sprang fram för att hjälpa , någon började filma ...

The content of the Wikipedia corpus is a community effort, which began some years ago, and is edited continually. It covers far-reaching topics, including those of the Swedish Gigaword corpus, and in addition, entertainment, art, politics and more. The sentences of the first seven lines of the content of the pre-processed version of the Wikipedia corpus are given below. It would be observed that it contains a bit of English words and the pre-processing script affected non-ascii characters. However, these issues were not serious enough to adversely affect the models generated, in this case, as the embedding system seems fairly robust to handle such noise.

> amager r en dansk i resund ns norra och v stra delar tillh r k penhamn medan vriga delar upptas av t rnby kommun och drag rs kommun amager har en yta p nine six two nine km och befolkningen uppg r till one nine six zero four seven personer one one two zero one eight en stor del av bebyggelsen har f rortspr gel men ven tskilliga innerstadskvarter finns i k penhamn samt i drag r p den stra delen av n finns kastrups flygplats amager r delvis en konstgjord delvis en naturlig s dan n r mycket l g och vissa delar ligger under havsytan framf r allt det genom f rd mning.

[11] created the Swedish analogy test set, which is similar to the Google analogy test set by [6]. This was because there was no existing analogy test set to evaluate Swedish embeddings [12, 13]. The analogy set has two main sections and their corresponding subsections: the semantic & syntactic sections. Two native speakers proof-read the analogy set for any possible issues (with percentage agreement of 98.93% between them), after valuable comments from the reviewers of this paper. It is noteworthy that some

words can have two or more possible related words. For example, based on the dictionary, the Swedish word *man* can be related to *kvinna* and *dam* in very similar ways. Four examples from the *gram2-opposite* sub-section of the syntactic section are:

medveten omedveten lycklig olycklig
medveten omedveten artig oartig
medveten omedveten härlig ohärlig
medveten omedveten bekväm obekväm

[9] correctly suggest there are problems with word similarity tasks for intrinsic evaluation of embeddings. One of the problems is overfitting, which large datasets (like the analogy set in this work) tend to alleviate [2]. In order to have a definite evaluation of embeddings, it's important to conduct experiments on relevant downstream tasks [9, 14, 15, 8].

3 Methodology

Table 1 gives the meta-data of the two corpora used. The Gigaword corpus was generated as described by [4] while the Wikipedia corpus was pre-processed using the recommended script by [16]. This script returned all text as lowercase and does not always retain non-asci characters. This created noise in the corpus, which may not necessarily be harmful, as it has been shown in a recent work that diacritics can adversely affect performance of embeddings unlike their normalized versions [17]. A portion of the pre-processed text (given in the previous section) was also tested for coherence on Google Translate and the English translation returned was meaningful, despite the noise. Hence, the noise issue was not serious enough to adversely affect the models generated in this case, as the embedding system seems fairly robust to handle such noise.

Meta-data	**Gigaword**	**Wikipedia**
Size	5.9G	4.2G
Tokens	1.08B	767M
Vocabulary	1.91M	1.21M
Year	2016	2019

Table 1: Meta-data for both Swedish Corpora

The authors made use of the fastText C++ library (with default hyper-parameters, except where mentioned) by [16] to generate 8 word2vec models and 8 subword models from each corpus, based on the optimal hyper-parameter combinations demonstrated by [10]. Each model was intrinsically evaluated using the new Swedish analogy test set by [11] in a Python-gensim program [18]. The hyper-parameters tuned are window size (4 & 8), neural network architecture (skipgram & continuous bag of words(CBoW)) and loss

(heirarchical softmax and negative sampling). The subword models used lower & upper character n-gram values of 3 & 6, respectively.

Although each model in the first set of experiments, with default (starting) learning rate (LR) of 0.05, was run twice and average analogy score calculated, it would have been more adequate to calculate averages over more runs per model and conduct statistical significance tests. Nonetheless, the statistical significance tests can be conducted for the downstream tasks, which usually are the key tests for the performance of these embeddings. It should also be noted that deviation from the mean of each model performance for their corresponding two runs is minimal. Due to the observation of one model (of Gigaword-CBoW-hierarchical softmax) failing (with *Encountered NaN* error) when using the default LR of 0.05, another set of experiments with the LR of 0.01 was conducted but with single run per model, due to time constraint.

4 Results & Discussion

Table 2 gives mean analogy scores for LR 0.05 of embeddings for the two corpora and table 3 for LR of 0.01. It will be observed that the skipgram-negative sampling combination for both corpora for word2vec and subword models performed best in both tables, except one in table 3, confirming what is known from previous research [6, 10, 11]. From table 2, the highest score is 60.38%, belonging to the word2vec embedding of the Wikipedia corpus. The lowest score is 2.59%, belonging to the CBoW-hierarchical softmax, subword embedding of the Gigaword corpus. The highest score in table 3 also belongs to the Wikipedia word2vec model. Among the 8 embeddings in the word2vec category in table 2, there are 6 Wikipedia embeddings with greater scores than the Gigaword while among the subword, there are 5 Wikipedia embeddings with greater scores. Nearest neighbour qualitative evaluation of the embeddings for a randomly selected word is given in table 4.

	Skipgram (s1)				**CBoW (s0)**			
	H. S. (h1)		**N. S. (h0)**		**H. S. (h1)**		**N. S. (h0)**	
window (w)	**4**	**8**	**4**	**8**	**4**	**8**	**4**	**8**
Word2Vec %								
Wikipedia	47.02	44.09	**60.38**	60.38	29.09	30.09	54.39	56.81
Gigaword	40.26	44.23	**55.79**	55.21	26.23	27.82	55.2	55.81
Subword %								
Wikipedia	46.65	45.8	**56.51**	56.36	28.07	24.95	38.26	35.92
Gigaword	41.37	44.7	**58.31**	56.28	2.59	-	46.81	46.39

Table 2: Mean Analogy Scores for Swedish Gigaword & Wikipedia Corpora with LR=0.05

	Skipgram (s1)				CBoW (s0)			
	H. S. (h1)		N. S. (h0)		H. S. (h1)		N. S. (h0)	
window (w)	**4**	**8**	**4**	**8**	**4**	**8**	**4**	**8**
Word2Vec %								
Wikipedia	48.92	49.01	51.71	**53.48**	32.36	33.92	47.05	49.76
Gigaword	39.12	43.06	48.32	**49.96**	28.89	31.19	44.91	48.02
Subword %								
Wikipedia	45.16	**46.82**	35.91	43.26	22.36	21.1	14.31	14.45
Gigaword	39.13	43.65	45.51	**49.1**	31.67	35.07	28.34	28.38

Table 3: Analogy Scores for Swedish Gigaword & Wikipedia Corpora with LR=0.01

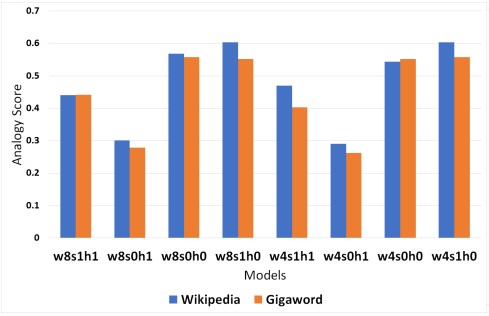

Figure 1: Word2Vec Mean Scores, LR:0.05

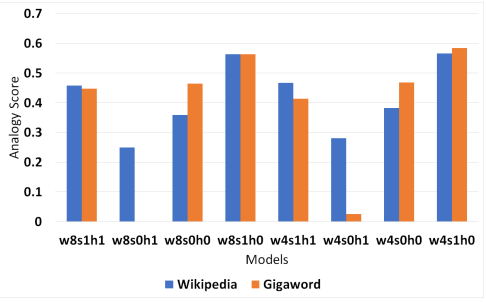

Figure 2: Subword Mean Scores, LR:0.05

Nearest Neighbor	Result
Wiki: syster	systerdotter (0.8521), systern (0.8359), ..
Gigaword: syster	systerdotter (0.8321), systerdottern (0.8021), ..

Table 4: Example qualitative assessment of Swedish subword w4s1h0 models

We hypothesize that the general performance difference observed between the embeddings of the two corpora may be due to a) the advantage of wider domain coverage (or corpus balance in topics) of the Wikipedia corpus - which is the most plausible reason, b) the small noise in the Wikipedia corpus or c) the combination of both earlier reasons.

Since it's preferable to have more than one criterion for the difference between the two corpora, future work will focus, particularly, downstream tasks to confirm this [9, 8]. Implementation without using the pre-processing script by [16] on the original Wikipedia corpus will also be attempted.

5 Conclusion

This work has shown that better performance results from differently sourced corpora of the same language can be based on reasons besides larger data size. Simply relying on larger corpus size for performance may be disappointing. The Wikipedia corpus showed better performance in analogy tests compared to the Gigaword corpus. Broad coverage of topics in a corpus seems important for better embeddings and noise, though generally harmful, may be helpful in certain instances. Future work will include other tests and downstream tasks for confirmation.

6 Acknowledgement

The authors wish to thank the anonymous reviewers for their valuable contributions and the very useful inputs from Carl Borngrund and Karl Ekström, who proof-read the analogy set. The work on this project is partially funded by Vinnova under the project number 2019-02996 "Sprkmodeller fr svenska myndigheter".

References

[1] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Conversational systems in machine learning from the point of view of the philosophy of science—using alime chat and related studies," *Philosophies*, vol. 4, no. 3, p. 41, 2019.

[2] E. Stevens, L. Antiga, and T. Viehmann, *Deep Learning with PyTorch.* Manning, 2020.

[3] M. T. Hagan, H. B. Demuth, and M. Beale, *Neural network design.* PWS Publishing Co., 1997.

[4] S. Rødven Eide, N. Tahmasebi, and L. Borin, "The swedish culturomics gigaword corpus: A one billion word swedish reference dataset for nlp," 2016.

[5] Wikipedia, "Swedish wikipedia multistream articles," 2019.

[6] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[7] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[8] A. Gatt and E. Krahmer, "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation," *Journal of Artificial Intelligence Research*, vol. 61, pp. 65–170, 2018.

[9] M. Faruqui, Y. Tsvetkov, P. Rastogi, and C. Dyer, "Problems with evaluation of word embeddings using word similarity tasks," *arXiv preprint arXiv:1605.02276*, 2016.

[10] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Word2vec: Optimal hyper-parameters and their impact on nlp downstream tasks," *arXiv preprint arXiv:2003.11645*, 2020.

[11] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Exploring swedish & english fasttext embeddings with the transformer," *arXiv preprint arXiv:2007.16007*, 2020.

[12] P. Fallgren, J. Segeblad, and M. Kuhlmann, "Towards a standard dataset of swedish word vectors," in *Sixth Swedish Language Technology Conference (SLTC), Umeå 17-18 nov 2016*, 2016.

[13] R. Précenth, "Word embeddings and gender stereotypes in swedish and english," 2019.

[14] M. Faruqui and C. Dyer, "Improving vector space word representations using multilingual correlation," in *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 462–471, 2014.

[15] A. Lu, W. Wang, M. Bansal, K. Gimpel, and K. Livescu, "Deep multilingual correlation for improved word embeddings," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 250–256, 2015.

[16] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning word vectors for 157 languages," *arXiv preprint arXiv:1802.06893*, 2018.

[17] T. P. Adewumi, F. Liwicki, and M. Liwicki, "The challenge of diacritics in yoruba embeddings," *arXiv preprint arXiv:2011.07605*, 2020.

[18] R. Řehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, (Valletta, Malta), pp. 45–50, ELRA, May 2010.

Paper D

The Challenge of Diacritics in Yorùbá Embeddings

Authors:
Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Reformatted version of paper accepted at:
ML4D Workshop at 34th Conference on Neural Information Processing Systems (NeurIPS 2020)

© 2020, The Publisher, Reprinted with permission.

The Challenge of Diacritics in Yorùbá Embeddings

Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Abstract

The major contributions of this work include the empirical establishment of a better performance for Yorùbá embeddings from undiacritized (normalized) dataset and provision of new analogy sets for evaluation. The Yorùbá language, being a tonal language, utilizes diacritics (tonal marks) in written form. We show that this affects embedding performance by creating embeddings from exactly the same Wikipedia dataset but with the second one normalized to be undiacritized. We further compare average intrinsic performance with two other work (using analogy test set & WordSim) and we obtain the best performance in WordSim and corresponding Spearman correlation.

1 Introduction

The Yorùbá language is spoken by about 40 million people in West Africa and around the world [1]. Of the various dialects around, the standard Yorùbá language (pioneered by Bishop Ajayi Crowther) is the focus of this paper. Standard Yorùbá orthography uses largely the Latin alphabet and is the widely spoken dialect among the educated [2]. Yorùbá has 25 letters in its alphabet, though counting the 5 nasal vowels makes it 30 [1, 3]. Being a tonal language, 3 diacritics are used on vowels based on syllables per word: depression tone (grave), optional mid tone and elevation tone (acute) [4]. Besides these differences between the English and the Yorùbá languages, Yorùbá has no gender identification for verbs or pronouns [5]. Yorùbá verb tenses are usually determined within context and remain mostly the same in spelling and tone [6, 7].

The research question we address in this work is "Do diacritics affect the performance of Yorùbá embeddings and in what way?" This is because it has been observed by [8] that web-search without diacritics produced more relevant results than search-words containing them, while evaluating four popular search engines. He also found out that the effectiveness of two of the search engines were adversely affected with diacritics. Thus, the objectives in this work include providing optimal Yorùbá embeddings and creating new analogy test set to evaluate the embeddings. Optimal hyper-parameter combination for the embeddings were chosen based on the work by [9, 10]. The heavily pre-processed (cleaned) Wikipedia dataset and the new analogy test set will provide valuable contributions to the natural language processing (NLP) community for the Yorùbá language, a low-resource language. The rest of this paper include the related work, the methodology, the results & discussion and the conclusion sections.

2 Related work

Initial effort by Ajayi Crowther to document Yorùbá barely had tonal marks [11]. In fact, early dictionary by [4] had minimal diacritics compared to the modern Yorùbá dictionary by [12]. This implies the language has been evolving and usage or discernment of diacritics between then and now is different. Revised efforts, later, standardized the diacritics and afforded others the opportunity to expand the work [3, 13]. For example, the word *abandon* in the [4] dictionary is *kọ-silẹ* while it is *kọ̀-sílẹ̀* in the modern Lexilogos dictionary[1] and that by [12].

Absence of diacritics made contextual semantics of words, probably, more important back then than they are today, given that some words with the same spelling can have different meanings, depending on the context. Even the English language has words which are spelled the same way but pronounced differently and have different meanings (homographs), exposed by context, e.g. *lead, row* or *fair*. Given the relative challenge of producing Yorùbá diacritics among some users, the versions without diacritics or partial diacritics have been increasing [8, 3, 13]. This has led some to push for the normalization (restricting diacritized letters to their base versions) of the Yorùbá language, especially in electronic media [8]. This attempt may also lead to canonicalization of Yorùbá text, through the relationship between diacritized and undiacritized words that will be established.

Other researchers, like [3] argue that diacritic restoration is a necessity. However, their own research showed the possible challenge for beginners of adding diacritics when the corpus they utilized had roughly the same percentage for the 3 diacritic marks [3]. Yorùbá diacritic restoration is being undertaken by some researchers from word-level, syllable-level or character-level restoration and some of the methods for automatic diacritization utilize Machine Learning (ML) methods [3].

Word embeddings have shortcomings, such as displaying biases in the data they are trained on [14]. However, they can be very useful for practical NLP applications. For example, subword representations have proven to be helpful when dealing with out-of-vocabulary (OOV) words and [15] used word embeddings to guide the parsing of OOV words in their work on meaning representation for robots. Intrinsic tests, in the form of word similarity or analogy tests, despite their weaknesses, have been shown to reveal meaningful relations among words in embeddings, given the relationship among words in context [16, 17]. It is inappropriate to assume such intrinsic tests are sufficient in themselves, just as it is inappropriate to assume one particular extrinsic (downstream) test is sufficient to generalise the performance of embeddings on all NLP tasks [18, 19, 9, 20].

3 Methodology

Three Yorùbá training datasets were used in this work. They include the cleaned 2020 Yorùbá Wikipedia dump containing diacritics to different levels across articles [21], a

[1] www.lexilogos.com/english/yoruba_dictionary.htm

normalized (undiacritized) version of it and the largest, diacritized data used by [22]. The original Yorùbá Wikipedia dump has a lot of vulgar content, in addition to English, French & other language content. Manual cleaning brought the file size down to 182MB from 1.2GB, after using a Python script to remove much of the HTML tags, from the initial raw size of 1.7GB. Using the recommended script by [23] to preprocess the original dataset did not work as intended, as it retained all the English & foreign content and removed characters with diacritics from the Yorùbá parts. An excerpt from the cleaned Wikipedia data, discussing about the planet Jupiter, is given below:

> Awo osan ati brown inu isujo Júpítérì wa lati iwusoke awon adapo ti won unyi awo wọn pada nigba ti wọn ba dojuko imọle [[ultraviolet]] lati ọdọ Orun. Ohun to wa ninu awon adapọ wọnyi ko daju, botilejepe fosforu, sulfur tabi boya [[hydrocarbon—haidrokarbon]] ni wọn je gbigbagbọ pe wọn je.

The authors created two analogy test sets: one with diacritics and an exact copy without diacritics. However, all results reported in the next section were for the standard diacritic versions of the analogy and WordSim sets. The results based on the undiacritized WordSim set for both Wiki versions were poorer than what is reported in the next section but the undiacritized Wiki version still gave better results than the diacritized against that set. Creating the analogy sets (containing over 4,000 samples each) was challenging for some of the sections in the original Google version by [16]. For example, in the *capital-common-countries* sub-section of the semantic section, getting consistent representations of some countries, like *Germany*, is difficult, as it is translated as *Jẹmani* by some or *Jamani* by others. A very useful resource is Lexilogos, which translates from English to Yorùbá and, importantly, displays a number of contextual references where the translation is used in Yorùbá texts. The analogy sets are smaller versions of the original, with 5 sub-sections in the semantic section and only 2 sub-sections in the syntactic section. All datasetsa and relevant code used are available for reproducibility of these experiments.[2] Four samples from the *gram2-opposite* of the diacritized version are given below:

> wá lọ àgbà ọdọ
> wá lọ òwúrọ ìrọlẹ
> wá lọ ọtá ọ̀rẹ́
> wá lọ nlá kékeré

Two types of embedding (word2vec and subword) per dataset were created, using the combination: skipgram-negative sampling with window size 4. The minimum and maximum values for the character ngram are 3 and 6, respectivley, though the embedding by [23] used ngram size of 5. Each embedding creation and evaluation was run twice to take an average, as reported in the next section. A Python-gensim [24] program was used to conduct the evaluations after creating the embeddings with the original C++ implementation by [23]. The Yorùbá WordSim by [22] was also used for intrinsic

[2]https://github.com/tosingithub/ydesk

Table 1: Yorùbá word2vec embeddings intrinsic scores (%)

Data	Vocab	Analogy	WordSim	Spearman
Wiki	275,356	0.65	26.0	24.36
U_Wiki	269,915	**0.8**	**86.79**	**90**
C3	31,412	0.73	37.77	37.83

evaluation. This Yorùbá WordSim was based on the original English version by [25], containing a small set of 353 samples. However, the Yorùbá version had a few issues, which we corrected before applying it. For example, *television* is translated as *tẹlifósìọ̀nù* instead of *tẹlifíṣọ̀n*, in one instance, and the bird *crane* is translated as *otí-bráńdì* (brandy) instead of *wádòwádò*, according to the Yorùbá dictionary.

4 Results & discussion

Tables 1 & 2 show results from the experiments while table 4 gives nearest neighbor result for the random word *iya (mother or affliction, depending on the context or diacritics)*. Average results for embeddings from the 3 training datasets and the embedding by [23] are tabulated: Wiki, U_Wiki, C3 & CC, representing embeddings from the cleaned Wikipedia dump, its undiacritized (normalized) version, the diacritized data from [22] and the Common Crawl embedding by [23], respectively. Performance of the original, contaminated Wikipedia dump was poorer than the cleaned version reported here, hence, it was left out from the table. It can be observed from table 1 that the cleaned Wiki embedding have lower scores than the C3, despite the larger data size of the Wiki. This may be attributed to the remaining noise in the Wiki dataset. Inspite of this noise, the exact undiacritized version (U_Wiki) outperforms C3, giving the best WordSim score & corresponding Spearman correlation. This seems to show diacritized data affects Yorùbá embeddings. The negative effect of noise in the Wiki word2vec embedding seems to reduce in the subword version in table 2.

The best analogy score is given by the embedding from [23], though very small. The performance of the embeddings are much lower for analogy evaluations than their English counterparts as demonstrated by [9], though the comparison is not entirely justified, since different dataset sizes are involved. Other non-English work, however, show it's not unusual to get lower scores, depending, partly, on the idiosyncrasies of the languages involved [10, 26]. NLP downstream tasks, such as named entity recognition (NER), with significance tests, will be the definitive measure for the performance of these embeddings, and this is being considered for future work.

5 Conclusion

The Yorùbá language is a tonal language and performance in NLP is affected, depending on diacritics, as shown in this work. It appears it is advantageous normalizing diacritized

Table 2: Yorùbá subword embeddings intrinsic scores (%)

Data	Vocab	Analogy	WordSim	Spearman
Wiki	275,356	0	45.95	44.79
U_Wiki	269,915	0	72.65	60
C3	31,412	0.18	39.26	38.69
CC	151,125	4.87	16.02	9.66

Table 3: Example qualitative assessment of undiacritized word2vec model

Nearest Neighbor	**Result**
iya	AgnEs (0.693), Arnauld (0.6798), oloṣajuloṣ (0.678), Rabiatu (0.6249), Alhaja (0.6186),..

texts before working on them for NLP purposes, as they produce better intrinsic performance, generally. Our embeddings, based on normalized text, achieved better instrinsic performance than others tested. Future work will involve utilizing the embeddings in downstream tasks, such as NER, using state-of-the-art (SotA) architectures. Such downstream tasks will serve as the definitive measure for evaluating these embeddings. There's ongoing effort on the sizable NER dataset to achieve this.

Broader Impact

The broader impact of this paper is the insight it provides for NLP researchers in Yorùbá language with regards to the differences in performance, based on diacritics. It provides 2 new analogy test sets for evaluating Yorùbá embeddings, depending on diacritics or the lack of it, and also provides an improved WordSim set. Furthermore, a heavily preprocessed Wikipedia dataset for training embeddings is provided, in the diacritized and undiacritized versions.

Acknowledgment

The work in this project is partially funded by Vinnova under the project number 2019-02996 "Språkmodeller för svenska myndigheter".

References

[1] K. J. Fakinlede, *Beginner's Yoruba*. Hippocrene Books, 2005.

[2] A. Bamgbose, *A grammar of Yoruba*, vol. 5. Cambridge University Press, 2000.

[3] F. O. Asahiah, O. A. Odejobi, and E. R. Adagunodo, "Restoring tone-marks in standard yorùbá electronic text: improved model," *Computer Science*, vol. 18, 2017.

[4] C. M. Society, *Dictionary of the Yoruba Language*. Church Missionary Society Bookshop, 1913.

[5] D. Nurse, S. Rose, and J. Hewson, "Verbal categories in niger-congo languages," 2010.

[6] T. Lamidi, "Tense & aspect in english & yoruba: Problem areas of yoruba learners of english," *Journal of the Linguistic Association of Nigeria Volume*, vol. 13, no. 2, pp. 349–358, 2010.

[7] A. H. Uwaezuoke and O. M. Ogunkeye, "A contrastive morphological analysis of tense formation in igbo and yoruba: implication on learners and teachers," *UJAH: Unizik Journal of Arts and Humanities*, vol. 18, no. 3, pp. 193–219, 2017.

[8] T. V. Asubiaro, "Effects of diacritics on web search engines' performance for retrieval of yoruba documents.," *Journal of Library & Information Studies*, vol. 12, no. 1, 2014.

[9] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Word2vec: Optimal hyper-parameters and their impact on nlp downstream tasks," *arXiv preprint arXiv:2003.11645*, 2020.

[10] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Exploring swedish & english fasttext embeddings with the transformer," *arXiv preprint arXiv:2007.16007*, 2020.

[11] T. J. Bowen, *Grammar and dictionary of the Yoruba language: with an introductory description of the country and people of Yoruba*, vol. 10. Smithsonian institution, 1858.

[12] P. Smith and A. Onayemi, "Yoruba dictionary," 2005.

[13] J. G. Fagborun, "Disparities in tonal and vowel representation: some practical problems in yoruba orthography," *Journal of West African Languages*, vol. 19, no. 2, pp. 74–92, 1989.

[14] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai, "Man is to computer programmer as woman is to homemaker? debiasing word embeddings," in *Advances in neural information processing systems*, pp. 4349–4357, 2016.

[15] J. Thomason, A. Padmakumar, J. Sinapov, N. Walker, Y. Jiang, H. Yedidsion, J. Hart, P. Stone, and R. Mooney, "Jointly improving parsing and perception for natural language commands through human-robot dialog," *Journal of Artificial Intelligence Research*, vol. 67, pp. 327–374, 2020.

[16] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[17] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

[18] A. Gatt and E. Krahmer, "Survey of the state of the art in natural language generation: Core tasks, applications and evaluation," *Journal of Artificial Intelligence Research*, vol. 61, pp. 65–170, 2018.

[19] M. Faruqui, Y. Tsvetkov, P. Rastogi, and C. Dyer, "Problems with evaluation of word embeddings using word similarity tasks," *arXiv preprint arXiv:1605.02276*, 2016.

[20] T. P. Adewumi, F. Liwicki, and M. Liwicki, "Corpora compared: The case of the swedish gigaword & wikipedia corpora," *arXiv preprint arXiv:2011.03281*, 2020.

[21] Wikipedia, "Yoruba wikipedia multistream articles," 2020.

[22] J. Alabi, K. Amponsah-Kaakyire, D. Adelani, and C. España-Bonet, "Massive vs. curated embeddings for low-resourced languages: the case of yorùbá and twi," in *Proceedings of The 12th Language Resources and Evaluation Conference*, pp. 2754–2762, 2020.

[23] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, "Learning word vectors for 157 languages," *arXiv preprint arXiv:1802.06893*, 2018.

[24] R. Řehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, (Valletta, Malta), pp. 45–50, ELRA, May 2010.

[25] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin, "Placing search in context: The concept revisited," in *Proceedings of the 10th international conference on World Wide Web*, pp. 406–414, 2001.

[26] M. Köper, C. Scheible, and S. S. im Walde, "Multilingual reliability and "semantic" structure of continuous word spaces," in *Proceedings of the 11th international conference on computational semantics*, pp. 40–45, 2015.

Paper E

Conversational Systems in Machine Learning from the Point of View of the Philosophy of Science — Using Alime Chat and Related Studies

Authors:
Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Reformatted version of paper originally published in:
Philosophies, 4(3)

© 2019, The Publisher, Reprinted with permission.

Conversational Systems in Machine Learning from the Point of View of the Philosophy of Science — Using Alime Chat and Related Studies

Tosin Adewumi, Foteini Liwicki and Marcus Liwicki

Abstract

This essay discusses current research efforts in conversational systems from the philosophy of science point of view and evaluates some conversational systems research activities from the standpoint of naturalism philosophical theory. Conversational systems or chatbots have advanced over the decades and now have become mainstream applications. They are software that users can communicate with, using natural language. Particular attention is given to the Alime Chat conversational system, already in industrial use, and the related research. The competitive nature of systems in production is a result of different researchers and developers trying to produce new conversational systems that can outperform previous or state-of-the-art systems. Different factors affect the quality of the conversational systems produced, and how one system is assessed as being better than another is a function of objectivity and of the relevant experimental results. This essay examines the research practices from, among others, Longino's view on objectivity and Popper's stand on falsification. Furthermore, the need for qualitative and large datasets is emphasized. This is in addition to the importance of the peer-review process in scientific publishing, as a means of developing, validating, or rejecting theories, claims, or methodologies in the research community. In conclusion, open data and open scientific discussion fora should become more prominent over the mere publication-focused trend.

1 Introduction

In this essay, the authors discuss conversational systems (also called chatbots) of natural language processing (NLP) in machine learning (ML), from the philosophy of science point of view. The authors' position on the theory of how science operates is one of naturalism [1]. Hence, the objective of this essay is to evaluate conversational systems' research activities in light of this philosophical theory. This theory of knowledge is similar to the precept and example of, the now defunct, logical empiricism, which viewed only verifiable statements as meaningful [1]. Understanding of the way the world functions or the theory that explains observations may influence what is perceived. Just as the scientific community holds on to certain assumptions alluded to by Kuhn [2], the conversational systems community is not exempt from these assumptions. The assumptions, central to naturalism, are a collection of beliefs and values, untested by the scientific

processes. They, however, give legitimacy to the scientific systems and set boundaries of investigations. One such basic assumption is that random sampling is representative for an entire population [3]. Possible benefits from this essay are that it summarizes improvements made in the science of developing conversational systems; and that it suggests that certain practices, such as the peer-review system and the use of qualitative, less biased, and large datasets, will bring further improvements.

Conversational systems are software systems that use natural language to communicate with users. This may be through written text or spoken dialogue [4]. The development of conversational systems began in the 1960s with Eliza being the product of such early studies [5]. This was a turning point in artificial intelligence (AI)—the imitation of human intelligence by software or hardware. AI is different from logical reasoning, problem-solving, or symbol's manipulation. However, some members of the AI community will agree that logic plays some role in the plethora of AI research areas [6]. Machine learning, which has become popular over the past few decades, is a subset of AI which is concerned with the learning of patterns for making predictions or performing specific tasks by using algorithms and statistical models without explicit programming [7]. The learning procedure takes place during training, with the aim of generalizing to 'unseen' data while avoiding overfitting (memorization) [8]. Natural language processing systems can be trained using text corpora (a large and structured set of texts) [9]. Examples of chatbots include Apple's Siri and Google Assistant.

Alime Chat is another chatbot developed by Alibaba researchers, mainly for Chinese [10]. It was developed for customer service operations at Alibaba[1] and can handle about 85% of the total customer service operation [10]. It is mainly a hybridized chatbot that leverages the capabilities of both information retrieval (IR) and machine learning generation models. Information retrieval and generation model approaches are categorized as data-driven because they rely mainly on data sources [11]. The latter synthesizes novel sentences, word by word, based on a dialogue history and persona (if included) [12, 13]. Meanwhile, the information retrieval approach retrieves stored information, such as documents, images, speech, and video, from repositories [9, 11, 14]. The reason for selecting Alime Chat research and its related studies is because they mark new trends in conversational systems' problem solving and many of them are being used in industry as well. Indeed, Alime Chat currently answers millions of customers' questions per day at Alibaba.

When a philosophy of science outlook regarding a given research subject is taken, there are at least two possibilities: One being to look at the research activities in the discipline being studied and evaluate the various philosophical theories proposed about the functioning of science and its epistemological status; the second being to adhere to a particular theory of how science operates and choose to evaluate the discipline's activities against the chosen philosophical theory; we chose the second approach. In the following, you will find the methodological issues section, the exposition of the chosen studies section, and the summary and conclusion section. The methodological issues section summarizes the approach and some metrics used in conversational systems research,

[1] alibaba.com

while the exposition section discusses some of the research activities from the point of view of the philosophy of science. Finally, the summary and conclusion section reiterates the main features of the discussion.

2 Methodological Issues

The methodology followed for gathering empirical data plays an important role. It must be unbiased (or impartial), as much as possible, and critical in its approach. Comparative studies, where performance of two or more systems are tested, are popular methods in conversational systems and they are usually based on experiments.

Various metrics (or measurements) exist in natural language processing. The BLEU score measures language translation success and was proposed by Papineni et al. [15]. It measures how closely machine translation is to standard human translation and how this correlates to human accuracy [15]. It is, however, reportedly not accurate when predicting single sentence human judgment, according to Lipton, Berkowitz, and Elkan [14] and therefore METEOR was introduced as an alternative. In addition, the GLEU score is an evaluation metric for sentence-level fluency [16].

Human or manual evaluation is considered a better metric than any other, since human understanding of what is produced is what is ultimately sought [13]. Despite the benefits of this type of evaluation, it has disadvantages: it is costly and subjective [17, 18]. It is costly in terms of resources (such as money and time) since the human subjects have to be recruited and trained before evaluation.

3 Exposition of the Chosen Studies

According to Thagard, when we can deduce statements, based on observation, from an occurrence, then a theory around such occurrence is verifiable [19]. For example, researchers in conversational systems, including Alime Chat, conduct several experiments and collect data by observation to make inferences [10, 11]. Inference refers to the process of drawing conclusions, sometimes done after a statistical analysis is carried out. Statistical analysis is the evaluation of data for the purpose of inference [3]. There are three main types of inference: deduction, induction, and abduction. What is inferred is necessarily true in deductive inferences, given true premises. Meanwhile, the nature of induction and abduction is one of non-necessary inference [20]. In a comparative study method, two or more systems' performances are assessed based on certain defined metrics (such as BLEU or GLEU) and the better or worse system is established from the outcome of several observations, as an average. Hence, though it is possible in some observations to find cases where a system with a low performance performs better than a system with a high performance, this is not sufficient enough to question the preeminence of the better system. Such a case can merely be seen as an anomaly. This is because one or a few out of many cases is not enough to invalidate a position, since many instances were conducted to arrive at an average.

Methods of inquiry require objectivity in their approach. Objectivity, whose value and attainability has been repeatedly criticized in the philosophy of science, is usually regarded as the basis of the authority of science or the reason for valuing science [21]. It prescribes that the components of science (such as methods and claims) should not be influenced by personal interests, community bias, or other similar factors [21]. Product objectivity and process objectivity are the two basic ways of understanding objectivity. Product objectivity is based on science's theories, experimental results (e.g., BLEU scores), observations, and similar products constituting accurate representations of the world [21, 22]. Process objectivity is multi-faceted and shows how science is objective to the point that the scientist's individual bias or contingent social values are not what science's processes and methods depend on [21]. An examination of the several conceptions of the ideal of objectivity is outside the scope of this essay. However, it has been argued that the facts of science are necessarily perspectival because of the involved apparatus and sociological factors [21]. Hence, given that full objectivity may not be deliverable, the conversational systems community plays a key role in describing what constitutes objectivity, which brings about trust in the science, as part of the social process. Indeed, Longino admitted that her analysis was not meant to be complete but to provide a starting framework from which the epistemologist (philosophers of the theory of knowledge) community could fill in further details [22].

Objectivity is a value which, as mentioned earlier, has been criticized extensively in the philosophy of science. Willingness to let the facts determine our beliefs, marks our objectivity. This is a position Longino does not seem to be averse to [22]. However, possible suspicion of what constitutes "the fact" from her submission, suggests that this needs to be carefully considered. For example, she suggests that the data used in a research experiment (which count as facts in that study) also need to be checked for reliability [22]. Hence, checking that the data has been interpreted by the authors in a subjective-free way is an important function in a peer-review process [22]. Furthermore, identification of possible institutional bias in the post-publication stage of a given idea was rightly identified by Longino [22]. This means that scientific publications should not be seen as the end. Attempts to reproduce experiments, subsequent use and modification by others are equally essential and can eventually compensate for institutional bias [22].

Conversational systems research makes use of the scientific method. The scientific method has process objectivity as its basis [21, 22]. As Longino pointed out, the scientific method is the use of non-arbitrary and non-subjective criteria for developing, accepting, and rejecting a scientific view [22]. Since objectivity itself may not be fully attainable, this has an impact on scientific methods, and again, makes the role played by the conversational systems community relevant to prescribing what constitutes the scientific method. This view is supported by Longino, who identified two shifts in perspective related to the scientific method, the second shift being made possible by refocusing on "science as practice". In her work, she proposes that this involves the subjection of hypotheses and the background assumptions to varieties of conceptual criticism [22]. Her point about objectivity of scientific methods being a function of both observational data and background assumptions lends credence to practices in the conversational systems

community [22]. Usually, the methods used in conducting experiments are provided for scrutiny, by researchers, to ensure their external and internal validity. Such information gives assurance to the conversational systems community about the objectivity of the results and the data used. Therefore, statistical analyses on such data can also be seen as objective. For example, Alime Chat researchers clearly stated the source of the data used, the architecture of the network, and the steps involved in producing the experiments [10]. This is also the case in a related study by Song et al. [11]. Furthermore, Longino observed that experiments based on unstable, quickly-evolving assumptions, lack objectivity [22]. Hence, observer effects, which may cause undue influence on research, are not objective. Methods employed in research should be a collection of social processes (such as the peer-review process for scientific publishing), as argued in [22]. This view is similar to Kuhn's position on the acceptance or rejection of a paradigm, which he argued should be a social process as much as a logical one [2].

In research on conversational systems, the type and size of data used for training influences the quality of the conversational systems created. For example, a small dataset utilized as an underlying corpus will produce poor performance when compared to a large dataset [9, 11]. Similarly, a biased dataset (either being a stereotyped dataset or a partial dataset) will be reflected in the performance of a conversational system, as was witnessed with Microsoft's chatbot Tay, which posted racist comments and conspiracy theories online after having been exposed to data of users who (intentionally or unintentionally) exploited the chatbot's sensitivity by posting many racist comments and conspiracy theories [23]. After valuable discussion with the anonymous reviewers of this essay, we should add that it is, in general, difficult to create an unbiased dataset. Indeed, for machine learning, a bias is typically needed to actually learn something. The most crucial issue, however, is to remove unwanted/harmful biases, such as racist, gendered, societal discriminatory, or hate-speech entries. Furthermore, an example for creating a less biased dataset (in the context of an insurance company) would be taking all inquiries (not only made in chats, but also by phone calls and physical visits) made by all customers and randomly selecting a subset of that. Public fora, such as conferences, workshops, and journals, provide avenues for criticism of research and its constituent parts. It is also through such avenues that shared standards can be learned and responses to criticism given. Despite concerns (such as unwarranted blocking of publications) regarding the peer-review process in scientific publishing, it is considered a very useful system for evaluating the objectivity of research methods and claims made in scientific papers [22]. It is a useful filter system that assesses whether an article conforms to generally agreed guidelines provided by the research community. The various articles on conversational systems cited in this essay were published in peer-review journals, which means they had been subject to some critical evaluation or criticism by members of the scientific community before being published.

In refuting conjectures, Popper was opposed to the procedure of inference as a result of many observations [24]. However, usually, claims made in conversational systems research are based on evidence from observations. This approach raises the concern of how many observations are sufficient to avoid refutation, as expressed by Popper. Furthermore,

Lipton categorically states that this approach cannot be taken as a proof of evidence [25]. Although abduction may be considered in a philosophical debate, the nature of the problem or debate plays an important part in its application, some even considering induction to be a special type of abduction [20]. Taking into account that we must be careful when concluding from empirical data, it is generally accepted that examples help in argument clarification and empirical confirmation and can increase the probability of the conclusion or claim. For example, Alime Chat researchers repeated 2136 tests in order to validate the obtained high performance of their system. Although Popper may have disagreed with this approach, the willingness of the conversational systems community to confirm or disconfirm their position, based on sufficient evidence, suggests that it is a reasonable approach. The willingness of the community to change, based on active research, is one of the scientific criteria alluded to by Thagard [19]. Lakatos may have approved this approach as the right one, since blind commitment is as serious a crime as any according to him [26]. Researchers in the area of conversational systems are not blindly committed to the claims or theories made, but are making strong efforts to ascertain the facts by reproducing experiments and are, in some cases, even advancing the field of research by trying out new methods. For instance, in determining if their hypothesis of a hybrid system was better, the Alime Chat researchers developed a new hybrid system and ran similar tests comparable to the old systems [10]. Song et al. similarly compared five architectures, including a baseline [11].

Confirmation by verification is not the only approach applicable in conversational systems, though this approach is sufficient for those who believe a theory is scientific only if it is verifiable [19]. The condition for refuting a claim can also be used. Popper states that in order for a claim or theory to be considered scientific, one should present a condition in which such a theory can be considered falsifiable or refutable [24, 26]. Such a test can be applied to some of the claims made in the conversational systems research society. For example, in order to compare Alime Chat with another chatbot in production, the researchers conducted 878 experiments on each of the chatbots [10]. In order to falsify their claim that Alime Chat was better, the researchers argued that the other chatbot had to win by conversing better (when answering questions, as evaluated by humans) in a majority number of times.

4 Summary and Conclusions

Standards and processes for conducting research in the area of conversational systems have been improved through the plethora of avenues created by the research community. In this essay, it has been shown that the mentioned research area uses scientific methods in developing, accepting, and rejecting proposed theories using rational and non-subjective criteria, as posited by Longino [22]. Full objectivity may not be realizable because of the apparatus of science and sociological factors (such as biases); however, the conversational systems community plays a key role in describing which components constitute the objectivity that brings trust. Furthermore, the importance of confirmation by verification was mentioned, as well as the use of falsification, as stated by Popper [24].

The process of improving the methodology employed in conversational systems research is lively and continual. This is especially important because we must be cautious when drawing conclusions from empirical data. Empirical confirmation, however, increases the probability of claims. The need for qualitative data as well as large amounts of data was pointed out in this essay. It is difficult to completely eliminate bias from datasets; however, efforts should be made to eliminate the presence of unwanted bias or stereotypes, which can negatively influence the performance of conversational systems. In addition, public fora, such as conferences, workshops, and journals, can provide the necessary avenues for criticism of the research in conversational systems, just as they do in other sciences.

Author Contributions

Conceptualization, Tosin P. Adewumi; Methodology, Tosin P. Adewumi; Refining of Concept and Methology: Foteini Liwicki and Marcus Liwicki; Investigation, Tosin P. Adewumi; Writing – Original Draft Preparation, Tosin P. Adewumi; Writing – Review & Editing, Foteini Liwicki and Marcus Liwicki; Supervision, Foteini Liwicki and Marcus Liwicki.

Funding

This research received no external funding.

Acknowledgment

The authors will like to appreciate and thank the Managing Editor and the anonymous reviewers of this essay for the very valuable feedback they provided.

Abbreviations

The following abbreviations are used in this manuscript:
AI: Artificial Intelligence
ML: Machine Learning
NLP: Natural Language Processing
IR: Information Retrieval

References

[1] R. Creath, "Logical empiricism," in *The Stanford Encyclopedia of Philosophy* (E. N. Zalta, ed.), Metaphysics Research Lab, Stanford University, fall 2017 ed., 2017.

[2] T. S. Kuhn, *The structure of scientific revolutions*. University of Chicago press, 2012.

[3] L. J. Kazmier, *Theory and Problems of Business Statistics*. McGraw-Hill, 2004.

[4] D. Burgan, "Dialogue systems and dialogue management," tech. rep., DST Group Edinburgh Edinburgh SA Australia, 2016.

[5] J. Weizenbaum, "Eliza—a computer program for the study of natural language communication between man and machine," *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966.

[6] R. Thomason, "Logic and artificial intelligence," in *The Stanford Encyclopedia of Philosophy* (E. N. Zalta, ed.), Metaphysics Research Lab, Stanford University, winter 2018 ed., 2018.

[7] N. Indurkhya and F. J. Damerau, *Handbook of natural language processing*, vol. 2. CRC Press, 2010.

[8] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[9] C. D. Manning, C. D. Manning, and H. Schütze, *Foundations of statistical natural language processing*. MIT press, 1999.

[10] M. Qiu, F.-L. Li, S. Wang, X. Gao, Y. Chen, W. Zhao, H. Chen, J. Huang, and W. Chu, "Alime chat: A sequence to sequence and rerank based chatbot engine," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 498–503, 2017.

[11] Y. Song, R. Yan, X. Li, D. Zhao, and M. Zhang, "Two are better than one: An ensemble of retrieval-and generation-based dialog systems," *arXiv preprint arXiv:1610.07149*, 2016.

[12] I. V. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[13] S. Zhang, E. Dinan, J. Urbanek, A. Szlam, D. Kiela, and J. Weston, "Personalizing dialogue agents: I have a dog, do you have pets too?," *arXiv preprint arXiv:1801.07243*, 2018.

[14] H. LI, "A short introduction to learning to rank," *IEICE Transactions on Information and Systems*, vol. E94-D, no. 10, pp. 1854–1862, 2011.

[15] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*, pp. 311–318, 2002.

[16] A. Mutton, M. Dras, S. Wan, and R. Dale, "Gleu: Automatic evaluation of sentence-level fluency," in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp. 344–351, 2007.

[17] A. Clark, C. Fox, and S. Lappin, *The handbook of computational linguistics and natural language processing*. John Wiley & Sons, 2013.

[18] A. Belz and E. Reiter, "Comparing automatic and human evaluation of nlg systems," in *11th Conference of the European Chapter of the Association for Computational Linguistics*, 2006.

[19] P. R. Thagard, "Why astrology is a pseudoscience," in *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association*, vol. 1978, pp. 223–234, Philosophy of Science Association, 1978.

[20] I. Douven, "Abduction," in *The Stanford Encyclopedia of Philosophy* (E. N. Zalta, ed.), Metaphysics Research Lab, Stanford University, summer 2017 ed., 2017.

[21] J. Reiss and J. Sprenger, "Scientific objectivity," in *The Stanford Encyclopedia of Philosophy* (E. N. Zalta, ed.), Metaphysics Research Lab, Stanford University, winter 2017 ed., 2017.

[22] H. Longino, *Values and objectivity*. na, 1998.

[23] V. Keselj, "Speech and language processing daniel jurafsky and james h. martin (stanford university and university of colorado at boulder) pearson prentice hall, 2009, xxxi+ 988 pp; hardbound, isbn 978-0-13-187321-6," 2009.

[24] K. Popper, *Conjectures and refutations: The growth of scientific knowledge*. routledge, 2014.

[25] P. Lipton, *Inference to the best explanation*. Routledge, 2003.

[26] I. Lakatos, "Science and pseudoscience," *Philosophical papers*, vol. 1, pp. 1–7, 1978.

Acronyms

CBoW continuous Bag-of-Words. 4, 7, 9, 15, 24

LSTM Long Short Term Memory Network. 11

MT Machine Translation. 7

MWE Multi-Word Expression. 7

NER Named Entity Recognition. v, 4, 10, 11, 13, 15–17, 23

NLG Natural Language Generation. 24

NLP Natural Language Processing. v, vii, 5, 7, 8, 10, 11, 19, 23, 24

NN neural network. 3–5, 8, 9, 11, 24

SA Sentiment Analysis. v, 10, 13–15, 23

SW Simple Wiki. 15

Department of Computer Science, Electrical and Space Engineering
Division of Embedded Intelligent Systems Lab

ISSN 1402-1757
ISBN 978-91-7790-810-4 (print)
ISBN 978-91-7790-811-1 (pdf)

Luleå University of Technology 2021

Tryck: Lenanders Grafiska, 136329

LULEÅ
UNIVERSITY
OF TECHNOLOGY