Extracting Lexical Stylistic Notions From Words Using LLMs

Cassie Huang, Seema Jahagirdar, Vaibhav Sahu, Luisa Silva
cassiehu@seas.upenn.edu, jseema@seas.upenn.edu
vaibhavs@seas.upenn.edu, luisafps@seas.upenn.edu

Abstract

Word embeddings from Large Language Models can be used in several different tasks, from sentiment analysis to textual style transfer and finding synonyms in other contexts. Our project goal is to use BERT word embeddings to characterize the complexity, formality, and figurativeness level of given words and documents. We do this using the implementation from Lyu et al. and expanding upon it. Our extensions include reducing anisotropy through k-means clustering, fine-tuning BERT models to perform the same task, and using other similarity metrics to perform the task. Our results show that reducing anisotropy through k-means clustering improves performance, and using other similarity metrics can improve performance without cosine similarity for some features, and fine-tuned BERT models can perform the same task at the document level.

1 Introduction

Word embeddings are adept at capturing the nuanced meanings and contexts of words through their position in a high-dimensional space. Unlike simpler feature vectors, they reflect semantic relationships and similarities by considering the co-occurrence or context of words in large text corpora. This allows them to better grasp and represent the subtleties of language, making them more effective for tasks like sentiment analysis, machine translation, and natural language understanding.

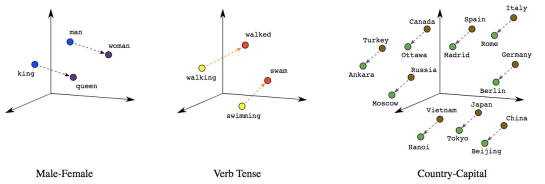


Figure 1: Linear Relationships between Words. Image from developers.google.com

Capturing lexical features using these high-dimensional spaces is an interesting area of research. Determining the directions that denote these lexical features can help us understand the structure of these spaces and can be extended to generate new meaningful samples from the latent space, denoting style transfer or debiasing the embeddings.

In order to extract lexical features and analyze the goodness of the said features, we use them to predict a dataset on features such as complexity and formality. We use accuracy as a metric to measure the goodness of these extracted features.

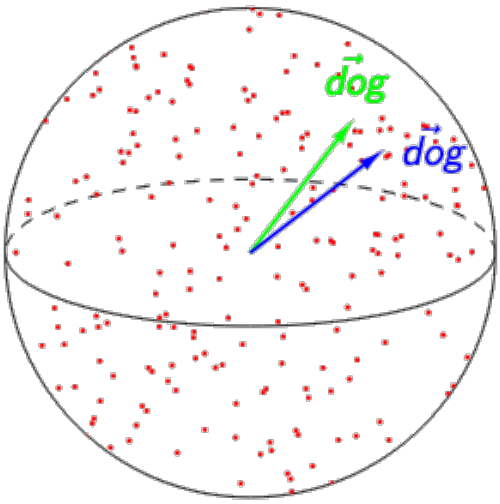
As shown in (Lyu et al., 2023), contextualized word representations do not perform as well as static word representations due to them being distributed in an anisotropic cone structure as shown in figure-2. This makes embeddings that are vastly different have high cosine similarities, which is not desirable for extracting meaningful features from these representations. While the paper discusses several ways to reduce anisotropy, we try to expand on this by first forming clusters of word representations and then using them to determine means, standard deviations, and principal components to remove anisotropy using methods like all-but-the-top and normalization.

We also explore the effects of these on custom similarity functions designed using a feed-forward neural network and compare them with cosine similarity.

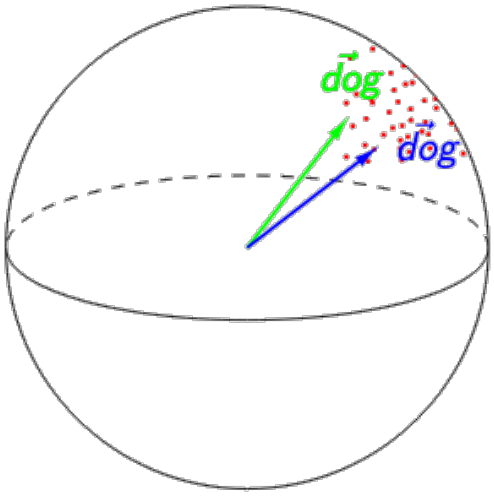
We finally, fine-tune these models to predict these lexical features on document-level tasks. Fine-tuning still remains to be the most powerful way to distinguish and classify these features. However, extracting meaningful lexical features at the document level becomes a seemingly uphill task.

2 Literature Review

The research paper (Lyu et al., 2023), "Representation of Lexical Stylistic Features in Language



(a) An isotropic embedding space



(b) An anisotropic embedding space

Figure 2: How the distribution of embeddings looks like in anisotropic and isotropic embedding spaces

Models' Embedding Space," explores how Language Models (LMs) encode lexical stylistic features like complexity, formality, and figurativeness. These features can be identified within the embedding space of pertained LMs by performing simple calculations. Vector Generation involves subtracting vectors representing semantic pairs that differ stylistically (e.g., Medical practitioner - doctor) to generate vectors representing complexity, formality, and figurativeness. The vectors are generated by averaging the differences in vectors across multiple pairs of words for each of the three features. Given a new word, phrase, or sentence, the cosine similarity between its vector representation and the complexity, formality, or figurativeness vector is computed. Two baselines are used: first, a majority baseline is employed, and then a frequency baseline is employed, where token frequency in the Google N-gram corpus determines the feature. It is observed that contextualized LMs outperform static embeddings in longer texts. Also, different embeddings like GloVe, fastText, BERT, mBERT, and RoBERTa show varying performance for different features and lengths of the text. For complexity and formality, max pooling performed better, whereas mean pooling was more efficient for figurativeness. The proposed method performs better than the baselines (majority and frequency) across static and contextualized LMs for capturing stylistic features. Anisotropy reduction methods, when applied to contextualized LMs, outperform static embeddings in all cases except formality on short texts.

The paper (Soler and Apidianaki, 2020), "BERT Knows Punta Cana is not just beautiful; it's gorgeous: Ranking Scalar Adjectives with Contextualized Representations," shows how BERT representations can encode rich information about the intensity of scalar adjectives (that describe a property of a noun at different degrees of intensity), which can be efficiently used for their ranking. The paper explores two methods of Scalar adjective ranking. The first is ranking with a reference point, which involves ranking all adjectives in a scale by taking the cosine similarity between their BERT representations and that of the most extreme adjective in the scale (which is the reference point). This is done across ten sentences retained for the scale and at every BERT layer, then the average cosine similarity values are used for ranking. The second method is ranking without specified boundaries, which addresses real-life scenarios where no concrete reference point is specified. It involves subtracting the representation of a mild-intensity adjective from that of an extreme adjective on the same scale. The resulting intensity vector is used to rank adjectives based on cosine similarity. The first method shows good performance, encoding intensity well, as indicated by moderate to high accuracy and correlation across three datasets, and the second method, especially with BERT embeddings, achieves high performance compared to baselines and previous work across multiple datasets.

Another paper we read was by (Bolukbasi et al., 2016), "Man is Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings".

This paper discusses the gender biases in Word2Vec embeddings and how to de-bias them. This paper explored how biased word embeddings can be by plotting embeddings of jobs along a gender axis, creating analogies, and asking crowd-workers whether the analogies contained stereotypes. The paper also explored biases in word embeddings by finding the gender subspace using the top principal component after using PCA. This gender subspace was then used for the debiasing algorithm. In the algorithm, we first find the gender subspace, then either Neutralize and Equalize or Soften. Results from the debiasing algorithm show that bias is reduced, but the essential properties of the original word embedding are preserved. Analogies also reflect less gender stereotypes. The paper concludes by saying that de-biasing word embeddings will not amplify biases in algorithms and can reduce societal biases.

One of the problems we faced in our project is that contextualized word embeddings have a high-dimensional space structure resembling that of an anisotropic cone. Words picked randomly with no resemblance can have a significant cosine similarity. While there are several ways to assess this problem, an elegant formulation of solving this problem was given in the paper (Rajaee and Pilehvar, 2021), 'A Cluster-based Approach for Improving Isotropy in Contextual Embedding Space' by Rajaee and Pilehvar. An earlier way of assessing anisotropy had been to remove the first few principal components of a word embedding matrix comprising an entire corpus of text or, in our case, the seed vectors for each task. The paper suggests that instead of globally removing anisotropy, the method of clustering word representations, before removing only the dominant principal components for each cluster separately, works much better.

3 Experimental Design

3.1 Data

Regarding our training/development/test data, we decided to use datasets: SimplePPDB for complexity tasks, Style-annotated PPD for formality tasks, and OneStopEnglish. SimplePPDB is used to extract complexity at the word level. It consists of .csv files (val.csv and test.csv) made up of word pairs and the label for which word is more simple. Style-annotate PPD is used to extract formality at the word level. It consists of .csv files (val.csv and test.csv) made of word pairs and the label for which word is more informal. Lastly, OneStopEnglish is used to extract complexity at the document level. It consists of text files separated by reading level. Refer to Table 1 for the dataset's size breakdown. To balance the label distribution in these datasets, the order of the two pieces of text in each pair is shuffled, and the label is assigned accordingly. Examples of all five datasets can be found in the appendix.

3.2 Evaluation Metric

When it comes to evaluation metrics, we decided to use accuracy. Accuracy is a metric that quantifies the overall correctness of predictions by calculating the ratio of correctly classified instances to the total number of instances. Accuracy is a good evaluation metric for a balanced dataset (like ours) because it measures the model's ability to correctly classify positive and negative instances without bias towards any specific class. Papers analyzed during our literature review, including "Representation of Lexical Stylistic Features in Language Models' Embedding Space" and "BERT Knows Punta Cana is not just beautiful, it's gorgeous: Ranking Scalar Adjectives with Contextualized Representations" use accuracy as their primary (and only) evaluation metric. Below, we can see the formula for accuracy, where TP are true positives, TN are true negatives, FP are false positives, and FN are true negatives.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

3.3 Simple Baseline

The simplest baselines we experimented with were the majority class and frequency baselines. The simple baseline experiments were done mainly on the SimplePPDB dataset. We used a dummy classier with a 'most frequent' strategy for the majority class baseline. The model uses the training data to figure out the most frequent label, and for every unseen input, it predicts that same most frequent label. For this method, we got an accuracy of 0.593 on the training set, 0.497 on the validation set, and 0.450 on the test set. The frequency baseline refers to the frequency occurrences of each token in the Google N-gram corpus (Brants, 2006). It considers more commonly used tokens simpler, more informal, and more straightforward, as frequency is a standard indicator of complexity and formality in prior research. This is because less common words tend to be more complex than those used

frequently. For this method, we got an accuracy of 0.547 on the training set, 0.555 on the validation set, and 0.807 on the test set.

Dataset	train	val	test
SimplePPDB	2943	814	1008
Style-annotated PPD	4597	367	367
OneStopEnglish	560	560	560

Table 1: Datasets sizes

4 Experimental Results

4.1 Published Baseline

The first published baseline from Lyu et al. includes 4 methods, all of which use an xlm-roberta-large model with pooling and using the 9th layer to get the embeddings. The first method uses abtt (All-but-the-top) with max pooling, the second uses abtt with mean pooling, the third uses max pooling (without abtt), and the fourth uses mean pooling (without abtt). Abtt is an anisotropy reduction technique. The anisotropy of contextualized LMs' representation space degrades the quality of the similarity estimates that can be drawn from it. Below, we can see the new representation x_{abtt} for an unseen word vector x, which is the result of eliminating the mean vector μ and the top k PCs (here $k = d/100$):

$$x_{abtt} = x - \mu - \sum_{i=1}^{k} (\mu_i^T x)\mu_i$$

The second published baseline we worked on uses logistic regression to identify the complexity of the text. Spacy's 'en_core_web_md' model and BERT model is leveraged to get text embeddings. The average of these token embeddings is computed to generate a single embedding for each text. These averages are then used to train the logistic regression model to find the complexity of words. The model learns a decision boundary in the feature space represented by the text embeddings that distinguish between simple and complex text.

See Table 2 for the test accuracies of the published baseline methods. Our implementation of the published baseline methods reaches the same level of accuracy as the original paper.

Method	Test Accuracy
abtt with max pooling	0.670
abtt with mean pooling	0.671
NO abtt with max pooling	0.859
NO abtt with mean pooling	0.838
Logistic regression	0.948

Table 2: Test accuracies of published baseline methods.

4.2 Extensions

4.2.1 Extension 1: Reduce Anisotropy using clustering

We have seen that while BERT embeddings can accurately model lexical features as vector directions, removing anisotropy from the embeddings helps the model achieve higher accuracies and model the features better. We have seen in (Lyu et al., 2023) a number of different methods to removing anisotropy, such as standardization, removing the top principle components through PCA, etc. Our first extension is to reduce anisotropy locally by using K-Means clustering and then applying reduction methods on each cluster. We have seen this work as demonstrated by (Rajaee and Pilehvar, 2021). We decided to combine both approaches and do a subsequent analysis.

Anisotropy Removal	**Feature**	**Accuracy**
abtt	complexity	0.635
abtt-optimally-clustered	complexity	0.835
normalized	complexity	0.785
normalized-clustered	complexity	0.85
abtt	formality	0.67
abtt-clustered	formality	0.689
abtt-optimally-clustered	formality	0.7

Table 3: Performance Summary

We use t-SNE to get an intuition of how the embedding space looks, as shown in figure-3. Intuitively, as the features and bi-valued, we see two essential clusters forming. This hints at the fact that our optimal number of clusters for removing anisotropy is 2.

In order to find the optimal number of clusters, we vary the number of clusters formed for anisotropy removal and track the performance in figure-4.

The same is done for the number of principal components removed from the embeddings, as shown in figure-5

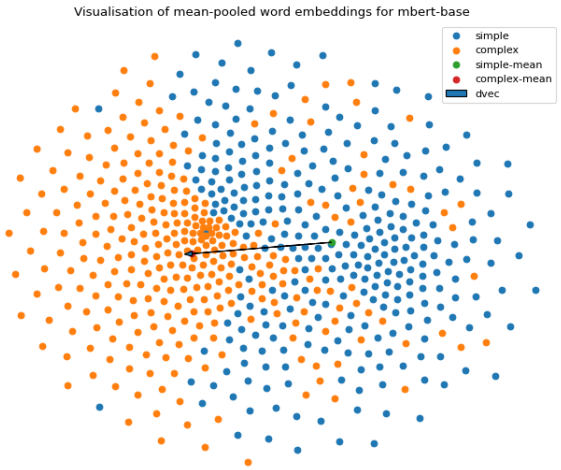
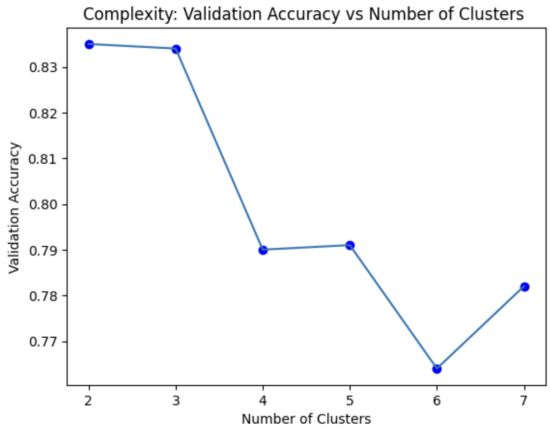
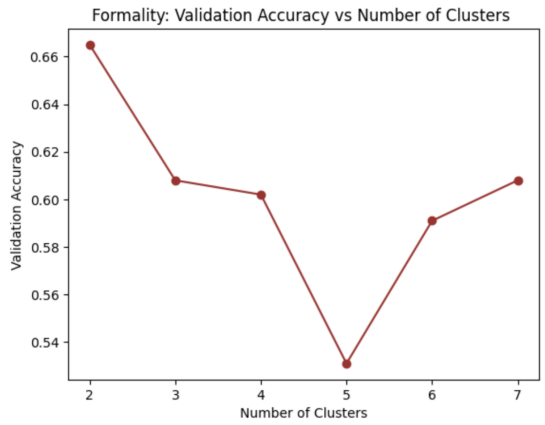


Figure 3: We use mbert-base embeddings and t-SNE to visualize embeddings



(a) Complexity Cluster Analysis

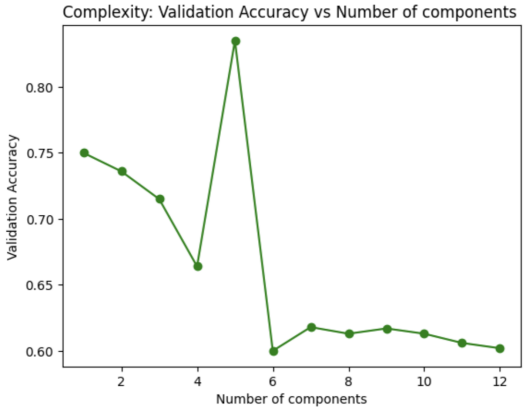


(b) Formality Cluster Analysis

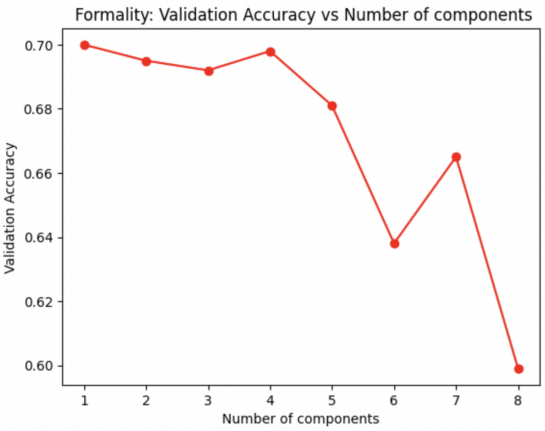
Figure 4: Cluster Analysis

4.2.2 Extension 2: Similarity Metrics

Another extension we have done is to change the similarity metric. In our published baseline, the embeddings are compared using cosine similarity. We decided to switch the metric by using other machine learning models to compare the embeddings. We chose this extension because our published baseline used cosine similarity to compare the similarity of the two embeddings, but we know that neural networks provide a more powerful framework to compare features and weigh them. Therefore, we wanted to see if we could improve upon the published baseline by using more powerful methods. Each model was given the feature vector and embeddings from both words as the input, and every model was evaluated using embeddings from the fourth layer of mBERT for complexity and the 1st layer of mbert for formality. We used a Neural Network, Random Forest, Gradient Boosted Classifier, AdaBoost Classifier, and SVMs using an RBF kernel and sigmoid kernel. Each model was given the feature vector and the embeddings of each word and was asked to predict which word is more simple for the complexity feature, and more informal for the formality feature. For the Neural Network, we created a network with 1 hidden layer with 3000 hidden units and trained



(a) Complexity Component Analysis



(b) Formality Component Analysis

Figure 5: Components Analysis

it using the Adam optimizer with a learning rate of 0.0001 for 40 epochs. Results from the best models can be found in table 4. From our results, we can see that cosine similarity performed better for complexity, but our similarity metrics performed better than cosine similarity for formality.

Similarity Metric	**Anisotropy Removal**	**Feature**	**Test Accuracy**
Sigmoid SVM	none	complexity	0.810
Neural Network	abtt	complexity	0.685
Neural Network	normalize	complexity	**0.839**
cosine similarity*	normalize	complexity	0.869
RBF SVM	none	formality	0.890
Neural Network	abtt	formality	0.861
RBF SVM	normalize	formality	**0.890**
cosine similarity*	abtt	formality	0.602
cosine similarity*	none	formality	0.52
Neural Net	clustered abtt	complexity	0.692
Neural Net	clustered normalize	complexity	0.825

Table 4: Results from varying the similarity metric. Rows with * are from the published baseline.

4.2.3 Extension 3: Fine-Tuning + Document Level

The baseline is from the paper OneStopEnglish corpus as in (Vajjala and Lučić, 2018). The baseline model utilized the Sequential Minimal Optimization (SMO) classifier with a linear kernel. To classify text complexity, the features employed included word n-grams, POS (Part-of-Speech) n-grams, character n-grams, syntactic production rules, dependency relations, as well as features typical in ARA research like lexical variation, psycholinguistic features, and discourse features. The results of the baseline showed that character n-grams performed the best among the generic features, achieving a relatively high accuracy rate of 77.25% for the text complexity classification task. When they combined multiple feature groups, they achieved an accuracy of 78.13%, highlighting the effectiveness of a combined feature set. The tables below show the results of the baseline.

Feature Group	Num. Feats.	Accuracy
Traditional	10	58.5%
Word	10	67.19&
Psycholinguistic	11	52.02%
LexVar, POS	29	72.48%
Syntactic Features	28	73.89%
Discourse Features	67	63.66%
Total	**155**	**78.13%**

Figure 6: Text Classification Results with generic features

Features	Accuracy
Word n-grams	61.38%
POS n-grams	67.37%
Char n-grams	**77.25%**
Syntactic Production Rules	54.67%
Dependency Relations	27.16%

Figure 7: Text Classification Results with specific linguistic complexity features

In this extension, we used three transformer models (Bert, Roberta, Distilled Bert) and fine tuned them using the OneStopEnglish corpus (document level). In this task, each document is classified as one of the three complexity levels, namely, elementary, intermediate, and advanced complexity, which represent increasing order of document complexity.

Both bert and distilled bert achieved an accuracy of 98.8%, a significant improvement over the baseline. Roberta achieved a test accuracy of 79.1%, very close to the baseline.

Parameters tuned are learning rate, epochs, batch size, and weight decay, which significantly enhanced the accuracy of the models.

- Increasing the number of epochs from 4 to 5 resulted in improved accuracy across all three models.
- Decreasing the learning rate from 0.00004 to

0.00005 also contributed to increased accuracy.

- Reducing the batch size from 16 to 8 notably improved accuracy, particularly in the Roberta and Distilled Bert models.

- The optimal settings achieved were: 5 epochs, Batch size of 8, and Lower weight decay: 0.0001 for the Distilled Bert model and 0.01 for the Bert and Roberta models.

The table below shows the accuracy obtained for different parameter settings.

Model	Val Acc	Test Acc	Learning rate	Epochs	Batch Size	Weight decay
Distilled Bert	0.965	0.977	0.00005	4	8	0.01
	0.835	0.849	0.00004	4	8	0.01
	0.800	0.779	0.00005	4	16	0.01
	0.941	0.965	0.00005	5	8	0.01
	0.965	**0.988**	**0.00005**	**5**	**8**	**0.0001**
Roberta	0.424	0.616	0.00005	4	8	0.01
	0.318	0.360	0.00004	4	8	0.01
	0.341	0.337	0.00005	4	16	0.01
	0.741	**0.791**	**0.00005**	**5**	**8**	**0.01**
	0.659	0.674	0.00005	5	8	0.0001
Bert	0.744	0.776	0.00005	4	8	0.01
	0.824	0.837	0.00004	4	8	0.01
	0.812	0.814	0.00005	4	16	0.01
	0.976	**0.988**	**0.00005**	**5**	**8**	**0.01**
	0.847	0.837	0.00005	4	8	0.0001

Figure 8: Performance of models with different parameter values

4.3 Error analysis

4.3.1 Extension 1

We look at three cases where feature vectors do not point in the right direction for complexity.

1. Correct-simple: standard
   Correct-complex: normal
   There is an obvious typo in the dataset. Unclean data can always be a reason for the under-performance of models.

2. Correct-simple: disturbs
   Correct-complex: impedes
   These words actually are a genuine error for our model. This can be attributed to the fact that dvec cannot capture the feature appropriately, as shown in figure-9. The direction for simple-to-complex is totally opposite to that of dvec. I think with better models and tuning for better layers to get embeddings from; we could fix such errors by having a better estimate for lexical feature vectors. Using anisotropy removal methods can also be a fix.

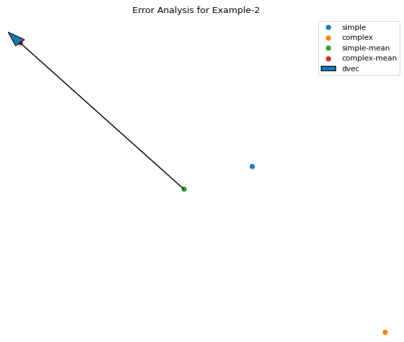


Figure 9: A Visual Illustration for example-2

3. Correct-simple: minimize
   Correct-complex: constrained
   Another reason for errors can be possibly due to subtle grammatical differences. These words are not in the same tense.

4.3.2 Extension 2

We look at three examples where our Neural Network does not predict the correct word as simple.

1. Correct-simple: alarmed
   Correct-complex: appal
   This is a typo in the dataset. Unclean data can always be a reason for the under-performance of models.

2. Correct-simple: liked
   Correct-complex: thought
   These two words do not have the same meaning, possibly contributing to the incorrect prediction.

3. Correct-simple: thurs
   Correct-complex: wif
   Neither of these examples are words, meaning the embeddings would not capture a correct word, leading to the error.

We also look at three examples where our RBF SVM does not predict the correct word/phrase as informal.

1. Correct-informal: online
   Correct-formal: available from
   Our model frequently predicts longer phrases as informal. This may be due to pooling the embeddings of multiple words.

2. Correct-informal: , the newspaper
   Correct-formal: the daily
   Punctuation was included in the data which may contribute to the incorrect predictions.

3. Correct-informal: it may not
   Correct-formal: she ca
   This example includes a typo and does not have the same subject, which may have led to the predictions.

4.3.3 Extension 3

We analyzed the Bert model, which performed well on the test data. However, it wrongly classified the below documents as simple, although they are complex.

1. Document 1 talks about social media and millennials. It uses nuanced vocabulary with terms like "intimacy overload" and phrases like "bucking the trend," which might pose a challenge due to their nuanced meanings. It has complex sentences like "A lot of my job is listening to people's lives all day, every day, and it started to feel so overwhelming to go on social media and see every single detail of everybody's life, including people that I don't really have a relationship with."

   The model wrongly classified this document as simple, as it might have struggled with parsing and analyzing the complexity hidden within the intricate sentence structures and nuanced vocabulary. It may not fully understand the depth of the reasoning and emotional context within these statements.

2. Document 2 talks about the message in a bottle. This document is complex and involves temporal references like "101 years after," "1946," and "at the age of 54," indicating specific historical timelines and events as well as precise historical details like "He died in 1946, six years before she was born."

   The model wrongly classifies this as simple as it might have struggled with understanding and contextualizing the historical and technical references, which could be more challenging to interpret.

3. Document 3 talks about clouds and air travel. It involves some technical terms like "cumulus cloud," "cumulonimbus," "vortices," and "inertial tracking". It includes scientific explanations about "light scattering" and "time travel" in the context of physics and atmospheric science.

   The model wrongly classified it as simple as it might struggle to recognize the complexity due to the presence of technical and scientific terms, as well as the detailed explanations about meteorology and aviation concepts.

5 Conclusion

In conclusion, we attempted to improve upon the published baseline from Lyu et al. in three extensions. Our first extension used clustering the reduce anisotropy. Our second extension involved using machine learning models as different similarity metrics instead of cosine similarity. Our third extension involved fine-tuning BERT to perform the same task at the document level. Clustering improved performance for normalized anisotropy reduction and abtt anisotropy reduction from our published baseline but did not meet state-of-the-art performance. This may be due to choosing the number of clusters and abtt performing worse overall. While using different similarity metrics improved upon some baselines, it did not surpass state-of-the-art performance. This may be due to hyperparameter tuning. Our third extension does extremely well compared to methods discussed in our published baseline but did not surpass state-of-the-art performance. This may be due to the dataset we have used. These reasons are all improvements to look into the future.

Acknowledgements

We would like to thank Veronica Lyu, Marianna Apidianaki and Chris Callison-Burch for the published baselines and code used for our extensions! Thank you to our TA Yu Feng for providing us feedback and brainstorming extensions for this project. We would also like to thank other TAs (Yufei Wang and Mona Gandhi) for brainstorming project ideas and baselines during their office hours. Finally, thank you Professor Mark Yatskar, for teaching a wonderful class this semester and his insights that helped us understand and finish our project! We would not have been able to complete this project without the help of everyone involved!

References

Tolga Bolukbasi, Kai-Wei Chang, James Zou, Venkatesh Saligrama, and Adam Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings.

Qing Lyu, Marianna Apidianaki, and Chris Callison-burch. 2023. Representation of lexical stylistic fea-

tures in language models' embedding space. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 370–387, Toronto, Canada. Association for Computational Linguistics.

Sara Rajaee and Mohammad Taher Pilehvar. 2021. A cluster-based approach for improving isotropy in contextual embedding space.

Aina Garí Soler and Marianna Apidianaki. 2020. Bert knows punta cana is not just beautiful, it's gorgeous: Ranking scalar adjectives with contextualised representations.

Sowmya Vajjala and Ivana Lučić. 2018. OneStopEnglish corpus: A new corpus for automatic readability assessment and text simplification. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 297–304, New Orleans, Louisiana. Association for Computational Linguistics.

A Appendix

A.1 Dataset examples

0	1	gold_simple
the wrath	rage	1
toys	playthings	0

Figure 10: SimplePPDB dataset example

0	1	gold_simple
The typhoon continued rapidly northeastward through the country and became an extratropical cyclone over northern HonshÅ `` a few hours after moving ashore .	"The typhoon continued moving rapidly northeastward through the country , and became an extratropical cyclone over northern HonshÅ `` ."	1

Figure 11: SimpleWikipedia dataset example

0	1	gold_informal
fellows	you guys	1
cut down	reducing	0

Figure 12: Style-annotated PPD dataset example

0	1	gold_literal
Villa Sylvia — relaxed holiday style	Villa Sylvia — free and easy holiday style	0

Figure 13: IMPLI dataset example

Intermediate

When you see the word Amazon, whats the first thing you think of the worlds biggest forest, the longest river or the largest internet shop and which do you think is most important?

These are questions in a debate about how to redraw the boundaries of the internet. Brazil and Peru have made objections to a bid made by the huge US e-commerce company for a prime new piece of cyberspace: .amazon.

The Seattle-based company has applied for its brand to be a top-level domain name (currently .com), but the South American governments argue this would prevent the use of this internet address for environmental protection, the promotion of indigenous rights and other public interest uses.

Together with many other disputed claims to names, including .patagonia, the issue goes directly to the heart of debates about the purpose and governance of the internet.

...

Figure 14: OneStopEnglish example text file

A.2 Extension 3 error analysis documents

Document 1

'That millennials rely heavily on technology is no secret. More than eight in ten say they sleep with a mobile phone by their bed, almost two thirds admit to texting while driving, one in five has posted a video of themselves online and three quarters have created a profile on a social networking site.

Compared to other generations, millennials are the most active on social media, according to a 2010 report, with 75% of them having created at least one social media account. In contrast, only 50% of Generation X, 30% of baby boomers and 6% of those aged 65 and older use social media. But there is a small percentage of millennials who don't use social media at all. Meet the millennials bucking the trend. \nCelan Beausoleil, 31, Oakland, California Beausoleil is a social worker and has had an "on and off, more off than on" relationship with Facebook. \nShe last deactivated her account in December 2015 after finding the amount of personal information shared by others "too heavy" to deal with on top of her work demands. \n "A lot of my job is listening to people's lives all day, every day and it started to feel so overwhelming to go on social media and see every single detail of everybody's life, including people that I don't really have a relationship with," she said. "It feels almost like intimacy overload." She added: "I'm holding

a lot in my work life for people and sometimes it felt like it was too heavy to do in my personal life also." \nBut Beausoleil does love the way social media connects the world in a truly unique way, citing it as one of her only reasons for staying on Facebook for as long as she did. "One thing I really liked about Facebook was that I could sit for hours and click on a friend and then click on one of their friends and one of their friends and one of their friends and literally end up on someone's Facebook page from the other side of the world," she said. "I used to do that all the time." \n "One day, I realized I'm spending so much time doing this. These little seconds add up. I wonder what it would be like if I didn't spend these seconds here and spent them doing something else. What if I was doing other things with these seconds? What would they become? Would I enjoy it?" \nMathias, who works for the Baltimore City government, had Facebook and Twitter accounts for years before deleting them both in November 2012. But he "quickly forgot that Facebook existed" after his impromptu decision to end his social media presence. He can still appreciate the benefits that come with having social media accounts, like when he met his girlfriend's friends for the first time and realized "humanizing 20 people you're meeting at a party" is much easier if you can connect their faces, hometowns and jobs to a photo later on. Or how easy it is to organize large events online. Mathias relies on friends for party invites and is sure there are times he "slips through the cracks". \nBut, now, he relishes the time that's freed up. He spends his lift rides and spare moments at work reading news articles and books rather than scrolling through a newsfeed. And with no friends' accounts to follow online, he has to "pick up the phone and call them", something he's come to "definitely enjoy". \nLauren Raskauskas, 22, Naples, Florida Raskauskas describes herself as a "pretty private" person. So social media, which can open you up to the scrutiny and analysis of others, is not that appealing to her. "I'm more privacy-minded and have concerns about giving out my data," said Raskauskas, who is currently looking for a job. \nShe recently deleted her Twitter account and deactivated her Facebook account two years ago after realizing she "didn't like everyone knowing what I was doing". But Raskauskas, who was late to the Facebook game because her "parents were really strict with technology", can see the positive sides of social media. When a friend of hers that she'd lost track of moved to Naples for a month, Raskauskas didn't even realize she was there until after she'd left, which the 22-year-old said "was a bummer". \nBut in the end, her privacy concerns outweighed any benefits social media could provide and she saw a definite upside when she went through a recent break-up. The last time a relationship of hers ended and she was online, it was not pleasant. \n "One time, I did break up with somebody while I was on Facebook and I was like' Oh my gosh, should I change my profile photo? Should I change my status?' And, this time, I don't have to worry about any of that because that kind of stuff is pretty hard," she said. \nRajagopalan, a student at Boston College, doesn't see any drawbacks to abstaining from social media. \nHe claims that he "hasn't seen any effect at this point". Even though classmates post about parties and events on Facebook, they make sure to send him a text message, too, he said. "Since I was young, I was always a step behind on that kind of thing so it never really mattered to me," he said. \nIn fact, the only time Rajagopalan made use of social media was when it was unavoidable: it was the only way to reach his new roommate at college. Before starting his first year at college, he signed up for his first, and only, social media account. He joined Facebook in order to contact his future roommate and talk about their plans for that year. \nMonths later, he still has the account but he admits: "I don't use it. I don't check it or anything like that." The most activity it sees is when his two sisters tag him in family photos. He has avoided social media accounts in all other situations, though he has felt the draw of Twitter. As a sports fan, he acknowledged that "it's where most of the news breaks out". But he refused to get an account, stating: "I don't really need one to read tweets" .\n'

Document 2

'Angela Erdmann never knew her grandfather. He died in 1946, six years before she was born. But, on Tuesday 8th April, 2014, she described the extraordinary moment when she received a message in a bottle, 101 years after he had lobbed it into the Baltic Sea. Thought to be the world's oldest message in a bottle, it was presented to Erdmann by the museum that is now exhibiting it in Germany. \n "It was very surprising," Erdmann, 62, said, recalling how she found out about the bottle. "A man stood at my door and told me he had post

from my grandfather. He then told me that a message in a bottle had been found and that the name that was on the card was that of my grandfather." Her visitor was a genealogical researcher who had managed to track her down in Berlin after the letter was given to the International Maritime Museum in the northern port city of Hamburg. \n The brown beer bottle, which had been in the water for 101 years, was found in the catch of Konrad Fischer, a fisherman, who had been out in the Baltic Sea off the northern city of Kiel. Holger von Neuhoff, curator for ocean and science at the museum, said this bottled message was the oldest he had come across. "There are documents that have been found without the bottle that are older and are in the museum," he said. "But, with the bottle and the document, this is certainly the oldest at the moment. It is in extremely good condition." \n Researchers believe Erdmann's grandfather, Richard Platz, threw the bottle in the sea while on a hike with a nature appreciation group in 1913. He was 20 years old at the time. \n Much of the postcard was indecipherable, although the address in Berlin on the front of the card was legible, as was the author's polite request that the note be sent by the finder to his home address. \n "He also included two stamps from that time that were also in the bottle, so the finder would not incur a cost," Erdmann said. "But he did not think it would take 101 years." \nShe said she was moved by the arrival of the message, although she had not known her grandfather because he died, at the age of 54, six years before she was born. \n "I knew very little about my grandfather, but I found out that he was a writer who was very open-minded, and believed in freedom and that everyone should respect each other," she said. "He did a lot for the young and later travelled with his wife and two daughters. It was wonderful because I could see where my roots came from." \nLike her grandfather, Erdmann said, she also liked culture and travelling around the world. She described herself as open-minded, too. "What he taught his two daughters, my mother taught me and I have then given to my sons," she said. Despite her joy at receiving the bottled message, she said that she hoped others would not repeat what her grandfather had done and throw bottles with messages into the sea. "Today, the sea is so full of so many bottles and rubbish that more shouldn't be thrown in there," she said. The message and the bottle will be on display at Hamburg's Maritime Museum until the beginning of May 2014, after which experts will attempt to decipher the rest of the text. It is not clear what will then happen to the bottle, but Erdmann hopes it will stay at the museum. \n "We want to make a few photos available to put with the bottle and give it a face, so visitors can see the young man who threw the bottle into the water," she said. '

Document 3

'1 Passing clouds \nOne of the pleasures of flying is seeing clouds close up. Even though they seem insubstantial they carry a considerable weight of water – around 500 tonnes in a small cumulus cloud. And water is denser than air. So why don't clouds fall out of the sky like rain? They do. But the droplets take a long time to sink. An average cloud would take a year to fall one metre. \n2 On cloud nine \nMost of us are happy to label clouds "fluffy ones" or "nasty black ones", but meteorologists identify more than 50 cloud types based on shape and altitude. These fit into categories given numbers from one to nine. Cloud nine is the vast, towering cumulonimbus, so to be "on cloud nine" implies being on top of the world. \n3 Around the rainbow \nThere's no better place to see a rainbow than from a plane. Rainbows are produced when sunlight hits raindrops. We see a bow because the Earth gets in the way, but, from a plane, a rainbow is a complete circle. When passing over clouds, the plane's shadow appears neatly in the centre of the effect. \n4 Mr blue sky \nSunlight is white, containing all the colours of the spectrum but, as it passes through air, some of the light is scattered when it interacts with the gas molecules. Blue light scatters more than the lower-energy colours, so the blue appears to come from the sky. \n5 There's life out there \nApart from clouds and other planes, we don't expect to see much directly outside a flying aircraft's window, but the air is seething with bacterial life – as many as 1,800 different types of bacteria have been detected over cities and they can reach twice the cruising height of a plane. \n6 Turbulence terror \nEven the most experienced flyer can be turned green by turbulence. The outcome can be anything from repeated bumping to sudden, dramatic plunges. The good news for nervous flyers is that no modern airliner has ever been brought down by turbulence. People have been injured and occasionally killed when they are not strapped in, or get struck by poorly secured luggage – but the plane is not going to be knocked out of the sky. \n7

In-flight radiation \nWhen body scanners were introduced at airports there were radiation scares but the level produced is the same as passengers receive in one minute of flight. The Earth is constantly bombarded by cosmic rays, natural radiation from space that has more impact at altitude. \n8 You can't cure jet lag \nThe world is divided into time zones. The result is that long-haul travel results in a difference between local time and your body's time, causing jet lag. However, its effects can be minimized by keeping food bland for 24 hours before travel, drinking plenty of fluids and living on your destination time from the moment you reach the aircraft. \n9 Supersonic 747s \nMany of us have travelled faster than sound. There are a number of jet streams in the upper atmosphere, notably on the journey from the US to Europe, where a temperature inversion causes a corridor of air to move as fast as 250mph. If an airliner with an airspeed of 550mph enters a jet stream, the result can be to fly at 800mph, above sound's 740mph. \n10 Flying through time \nTime zones provide an artificial journey through time – but special relativity means that a flight involves actual time travel. It's so minimal, though, that crossing the Atlantic weekly for 40 years would only move you 1/1,000th of a second into the future. \n11 Terrible tea \nDon't blame the cabin attendant if your tea isn't great. Water should be just under 100°C when it is poured on to tea leaves – but that isn't possible on a plane. It's impossible to get water beyond 90°C during flight – so choose coffee. \n12 I can't hear my food \nAirline food has a reputation for being bland and tasteless. Some of the problem may not be poor catering, though. A plane is a noisy environment and there is evidence that food loses some of its savour when we are exposed to loud noises. \n13 Needle in a haystack \nWith modern technology, it seems strange that Malaysian flight MH370 could disappear without a trace – yet, finding a missing aircraft is a needle- in-a-haystack problem. The plane knows its location, both from GPS and inertial tracking, but this information is not relayed elsewhere in real time. That would be perfectly possible. Ocean- going ships have had tracking since the 1980s – the limitation is not technology but a lack of legislation requiring it. \n14 Volcanic fallout \nAir travel can be cancelled by volcanic activity. Glass-like ash particles melt in the heat of the engine, then solidify on the rotors. A clear-skies policy in an ash cloud may be inconvenient – but the risks of ignoring the ash are clear. \n15 The wing myth \nFor many years, we taught the wrong explanation for the way wings keep planes in the air. In fact, almost all a plane's lift comes from Newton's Third Law of Motion. The wing is shaped to push air downwards. As the air is pushed down, the wing gets an equal and opposite push upwards, lifting the plane. \n16 Forget electric planes \nWhen we see ultra-light, experimental, electric planes, it's easy to assume there will soon be clean, green, electric airliners, but it won't happen any time soon. Aircraft fuel packs in a remarkable amount of energy. Batteries are much less efficient. To provide the same energy as a tonne of fuel would take 100 tonnes of batteries – and a 747 uses 150 to 200 tonnes of fuel. Unless battery technology is made vastly more efficient, electric airliners won't get off the ground. \n17 Beware the vortex \nPilots often wait a long time to get clearance. This is to allow the air to settle after a previous take-off, as a plane's wingtips generate vortices in the air, which can take two or three minutes to disperse. If the following aircraft set off immediately, the rapidly moving air would make the plane difficult to handle. The delay gives the air time to recover from the miniature whirlwinds caused by the preceding plane. \n18 The doors aren't locked \nIn practice, the doors on a plane don't need to be locked. If you watch an aircraft door being opened, it swings in an unusual way. It first has to be opened inwards before manoeuvring it out of the way. Once the plane has taken off, a significant pressure difference soon builds up between the inside of the plane and the outside. This differential forces the door into place. To open it, you would have to pull against the air pressure, well beyond the capabilities of human muscles.\n',

A.3 Code

The code can be found in the repo folder "Expanding-on-Lexical-Stylistic-Features-modified". While, most of the analysis is done in Jupyter notebooks.

A.3.1 Extension-1

Extension-1 involved modifying the entire model class, which is referred to as the "LexicalFeaturizer" class defined in the "models" directory in the repository code, to include more words from the training data to determine anisotropy removal parameters as well as cluster them and remove anisotropy from each clus-

ter. This involved making a new method, "compute_clustered_mean_std_pcs", for the model class, as well as modifying the methods to generate embeddings, dvecs, as well as predict them using the similarity method.

Analysis for milestone-1 comprises code in the notebooks "Milestone-3 - Analysis - complexity - final" and "Milestone-3 - Analysis - formality - final" for tuning for the number of clusters and PCs. The overall results are summarized in the notebook, "Milestone-3 - Extension-1 - final". Further analysis, like visualization, is done in "Visualize_embeddings" notebook and "Error_analysis" notebook.

A.3.2 Extension-2

Extension-2 involved taking code written in the "Expanding-on-Lexical-Stylistic-Features-modified" repository and generating embeddings for the words in the train, val and test set. Then we used jupyter notebooks to create the embeddings, load dvecs and generate machine learning models as different similarity metrics. For each different anisotropy reduction method, we used a separate notebook to generate the model and analyze the results.

For similarity metrics involving complexity, refer to notebooks "Milestone 4: Similarity with ABTT COMPLEXITY", "Milestone 4: Similarity with Clustered ABTT COMPLEXITY", "Milestone 4: Similarity with Normalization COMPLEXITY", "Milestone 4: Similarity with Clustered Normalization COMPLEXITY", and "Milestone 4: Similarity without ABTT COMPLEXITY".

For similarity metrics involving formality, refer to notebooks "Milestone 4: Similarity with ABTT FORMALITY", "Milestone 4: Similarity with Normalization FORMALITY", and "Milestone 4: Similarity without ABTT FORMALITY".

A.3.3 Extension-3

The entirety of extension-3 is explained in a notebook format in the "Three-labels-Finetuning-script" notebook. This involves fine-tuning the model as well as error analysis on incorrectly classified examples.