What's in a Gist? Towards an Unsupervised Gist Representation for Few-Shot Large Document Classification

Jaron Mar(✉) and Jiamou Liu

The University of Auckland, Auckland, New Zealand
{jaron.mar,jiamou.liu}@auckland.ac.nz

Abstract. The gist can be viewed as an abstract concept that represents only the quintessential meaning derived from a single or multiple sources of information. We live in an age where vast quantities of information are widely available and easily accessible. Identifying the gist contextualises information which facilitates the fast disambiguation and prediction of related concepts bringing about a set of natural relationships defined between information sources. In this paper, we investigate and introduce a novel unsupervised gist extraction and quantification framework that represents a computational form of the gist based on notions from fuzzy trace theory. To evaluate our purposed framework, we apply the gist to the task of semantic similarity, specifically to few-shot large document classification where documents on average have a large number of words. The results show our proposed gist representation can effectively capture the essential information from a text document while dramatically reducing the features used.

Keywords: Semantic representation · Few-shot learning · Unsupervised learning

1 Introduction

The gist can be viewed as an abstract concept that represents only the quintessential meaning derived from a single or multiple sources of information such as images or text. This brings into question how can an abstract concept such as the gist be computationally extracted and quantified? Due to this abstract nature, there does not exist a common formalism of the gist in psychology, computational linguistics or NLP. In computational linguistics the notion of gist appears in gist preservation for discourse [22] and gisting or summarisation [18]. Our view of the gist differs from these works in the sense that we view the gist as a very high level but low dimensional semantic representation of a text which can novelly be quantified as a real number known as the *gist score*. To evaluate the predictive capabilities of the gist score we experimentally applying our representation to few-shot document classification and compare

H. W. Lauw et al. (Eds.): PAKDD 2020, LNAI 12084, pp. 261–274, 2020.
https://doi.org/10.1007/978-3-030-47426-3_21

the results to existing classical and few-shot classifiers. In particular, we focus on the classification of large documents where documents on average are comprised of thousands of words. Standard datasets for document classification such as Yelp, IMDB, Reuters, etc. contain documents with only hundreds of words on average. As such, few works evaluate large document classification partly due to the fact that neural-based models have large memory requirements to train over documents with large numbers of words or are only trained using sections of the document. Intuition also tells us that extracting and quantifying the gist of a document should be more accurate in longer documents.

Few-shot or N-shot learning is a recent paradigm of learning originally applied in computer vision with high levels of success [7]. There has been recent interest in applying this methodology to NLP however the nature of language makes the complexity of few-shot learning more difficult. Many approaches to few-shot learning employ some form of transfer or meta-learning which requires supervised learning over related classes. However, in a real-world application where N-shot learning could be applied, it is also unlikely that suitable labelled training data of related classes will exist. Therefore, taking an unsupervised approach to few-shot learning is arguably more important and we show that few-shot learning in NLP can be approached in an unsupervised way by determining the semantic similarity based on our purposed method using the gist score. The fundamental difference in our approach compared to few-shot approaches in computer vision can be seen in the following problem formulations.

Typical Few-Shot Problem Formulation: Given a set of labelled training examples $\mathcal{X}^{train}$ with classes $\mathcal{Y}^{train}$. The goal is to create a model that acquires knowledge from the training examples such that the knowledge facilitates the prediction of the test set $\mathcal{X}^{test}$ using a few (N) labelled examples of each class in $\mathcal{Y}^{test}$ and classes in $\mathcal{Y}^{test}$ are disjoint but related to those in $\mathcal{Y}^{train}$.

Our Few-Shot Problem Formulation: In a more restrictive but more realistic scenario where given only the test set $\mathcal{X}^{test}$ to classify and N examples of each class from $\mathcal{Y}^{test}$ the goal is to predict the classes of $\mathcal{X}^{test}$ i.e. perform unsupervised N-shot learning.

Contributions and Paper Organisation: This paper presents two main contributions listed in the order they are described in the paper. (1) We propose an unsupervised gist representation that provides a link between psychology and computational linguistics which allows for the extraction and quantification of the gist from a text. This acts as a first step towards a possible computational model of FTT. (2) We apply the gist score to few-shot large document classification and experimentally show that our gist representation performs as well as centroid based similarity measures and better than traditional baseline algorithms while effectively only using a one-dimensional representation for each word. This has significant implications towards the dimensionality required for word embeddings by suggesting it is possible to train a greatly reduced embedding while still retaining high accuracy in downstream tasks.

2 Related Works

The concept of the gist is predominately found in cognitive and psychological contexts observing the manifestation of gist in human behaviour [15]. The concept of gist can also be found in artificial intelligence to explain norm emergence using gist information [10] and for summarising image information using gist descriptors [6], few works attempt to computationally quantify the gist in language. Many computational models were inspired by cognitive models which have been developed and studied in psychology. Our work is closely related to the psychological notion of fuzzy-trace theory (FTT), a well-founded theory that states that reasoning occurs on simple gists rather than exact details [4]. Underlying FTT are seven key principles, the first being gist extraction which is the task of reducing information to its essence. In FTT, it is hypothesised there are two cognitive systems of memory, the verbatim memory and the gist memory. The verbatim memory acts to retain detailed information whereas the gist memory acts to retain only the quintessential information. Therefore, assuming a sequential memory model the gist memory can be thought of as a compressed representation of the verbatim memory. In this paper, we explore a computational model for FTT that allows for the extraction and inference of the gist based on the underlying principles of FTT.

In NLP, gist extraction has previously been defined as a statistical problem where the goal is to infer $P(gist|words)$ using a generative model for language [9] where the gist is seen as a latent variable. In particular, latent Dirichlet allocation (LDA) [2] a generative topic model assumes that the distribution of the gist over topics is drawn from a Dirichlet distribution. The topic model presents a very structured representation that assumes there exists a strict predefined latent structure and dependencies on how the language was generated, our work differs from this as it assumes no such structure and no distribution over topics. Furthermore, the gist generated from these generative models have no direct application to other downstream tasks. In this paper, we apply our notion of the gist to document classification, the task of determining and assigning the classes of documents. This is an important and well-studied task in NLP with applications in information retrieval and spam identification with many different approaches from term frequency [12], state vector machines (SVM) [11], current state of the art neural-based methods [24] and recent few-shot learning approaches [8,23].

3 Gist Representation Framework

Intuitively, the gist of a text represents the singular most important semantic meaning that is representative of the overall text. We therefore define the *gist score* as a real number which encapsulates the above notion by condensing the semantic information of a text. The intuition behind such a compressed representation for the gist score is based on two psychological notions. The first notion being the *fuzzy-to-verbatim continua* [4] which states that the that people

encode multiple representations at varying levels of precision along the fuzzy-to-verbatim continua, this suggests that information can be encoded as a real number. The second notion is a phenomenon displayed in humans known as *fuzzy processing preference* [5] which states that to make decisions they will use the least precise gist representation which allows for faster inferencing. If a human were to judge the similarity between two vectors of multiple dimensions or two real numbers, at a glance a human can easily determine the difference between two numbers.

This section will outline the intricacies around the seemingly simple gist representation framework proposed in Algorithm 1 motivated by an example to produce the gist embeddings in Fig. 1 using two chapters, C029 (red) and C002 (orange) from a medical textbook and the zebra Wikipedia page (purple) from datasets used in Sect. 5.1.

Algorithm 1. Gist Extraction Framework, GistEmbedding($T, size$)

Input: Text T, Segmentation size
Output: 1-Dimensional gist embedding of each word

1: $T \leftarrow preprocess(T)$
2: $embeddingArray \leftarrow embed(T)$
3: $segments \leftarrow segment(embeddingSet, size)$
4: $embeddingArray \leftarrow centroid(segments)$
5: $gist \leftarrow reduce1Dim(embeddingArray)$

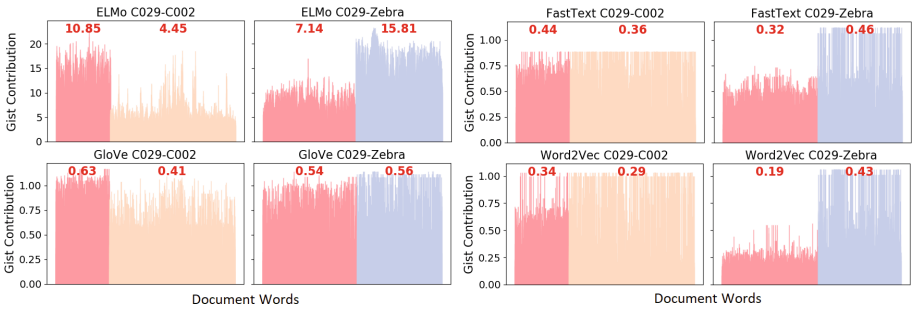

Fig. 1. Gist embeddings and scores with varying embedding methods for the verbatim memory. (Color figure online)

3.1 Modelling the Verbatim Memory via Semantic Embeddings

In FTT, verbatim representations are detailed and precise representations of complete and exact information. To model text as a verbatim representation, line 2 of the algorithm we use semantic embeddings, particularly word embeddings as it gives the highest level of granularity. In this paper we investigate popular word embedding models ELMo [20], FastText [3], GloVe [19] and Word2Vec [16] to determine which models are suitable for uncovering the gist using widely available pretrained models for each.

3.2 Verbatim to Gist Representation via Dimensionality Reduction

The gist only needs to represent the quintessential meaning, we aim to encapsulate only this essential semantic meaning by radically reducing the verbatim representation to one dimension using principal component analysis (PCA). Applying dimensionality reduction to embeddings is not uncommon and has been used as a post-processing step to reduce the size of the embeddings while still maintaining the quality of the embeddings when applied to downstream tasks [21] but the degree of reduction is not as radical.

It has long been known that the word frequency in language follows a power law distribution [25] which suggests that the embedding set generated from these words will also follow a power law distribution, especially in large documents. Given a centred word embedding set, a PCA to one dimension creates a linear projection such that the variance of the first principal component $\vec{p}$ is maximized for $\frac{1}{n}\sum_i^n(\vec{e_i}\cdot\vec{p})^2$ where e_i is the embedding for word i which in turn minimizes the mean squared error (MSE). Since PCA aims to minimize the MSE and the embedding set follows a power law distribution where a relatively small set of common words occur often, then intuitively p will be chosen such that the MSE is minimized for these common word embeddings. When p is chosen in this way the projected value for the common words will have a lower value as they contribute less to the variation on the first principal component. Conversely, the unique words that corroborate more closely to the gist will have a higher value in the resultant PCA. For example, when reducing the combined C002-C029 texts to 1-dimension via PCA using Word2Vec embeddings the words "patients", "physicians" and "clinicians" have the highest values in both the C002 and C029 segments. Figure 1 shows the result of a PCA on our example in which the global structure is preserved.

3.3 Quantifying and Extracting the Gist Score

Quantify the gist score from the gist embedding can be achieved simply by taking the average of the segment that corresponds to each document in the embedding as shown in bold red font in Fig. 1. To better understand what the average captures and how it can capture the gist, suppose we can treat the gist embeddings in Fig. 1 as values from a sample population that are drawn from a semantic distribution over words. Kernel density estimation (KDE), a non-parametric technique that estimates the probability density function (PDF) applied to the values of the gist embedding gives the probability of sampling a word or segment with a given gist contribution value. The KDE plot will uncover the underlying gist contribution distribution and thus the underlying semantic similarity of words from the original embedding set.

Evaluating the KDEs in Fig. 2 based on embeddings generated in Fig. 1 we see using ELMo embeddings it is obvious that the KDE based on C029-Zebra produces a bimodal distribution showing the existence of two classes, whereas C029-C002 is unimodal showing that these documents are highly similar. Therefore, when average and evaluating the distance been the gist scores the distance

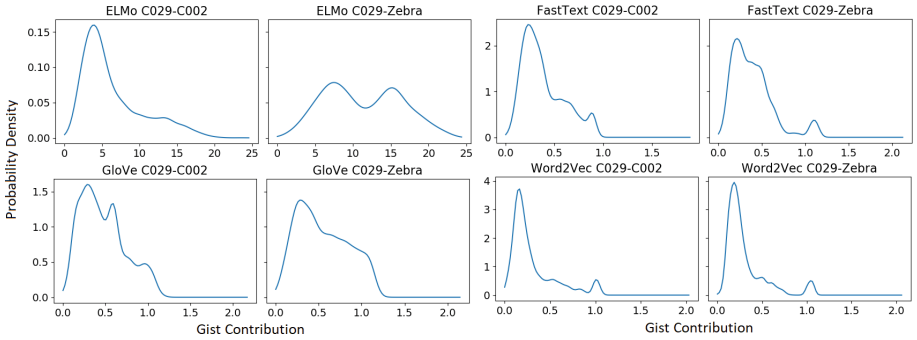

Fig. 2. Equivalent KDE plots with a Gaussian kernel and Scott's bandwidth selection for Fig. 1.

between C029-C002 and C029-Zebra should be lower because of the differences in the distributions, which is the case with values of 6.40 and 8.67. Comparing the gist embedding using Word2Vec and PCA we see that the distributions look almost identical due to the nature of having a single embedding for each word. The key difference is that the mode representing the highest values at $x = 1$ representing the unique words are spread across both documents in the C029-C002 gist embedding and only these values only occur in the zebra document in the C029-Zebra case as shown in Fig. 1 which gives the gist scores 0.05 and 0.24.

3.4 Uncovering the Underlying Gist via Segment Centroids

Producing a gist embedding over all words can uncover the gist by taking into account all the semantic information however, to capture the gist the importance of individual words is low and can introduce noise. A key principle of FTT states the gist can be encoded at different levels of fuzziness along the fuzzy-to-verbatim continua. Therefore, to investigate further fuzzy gist representations we introduce a method that approximates the gist embedding at a less granular level by taking gist embeddings over centroids of partitioned segments from the text.

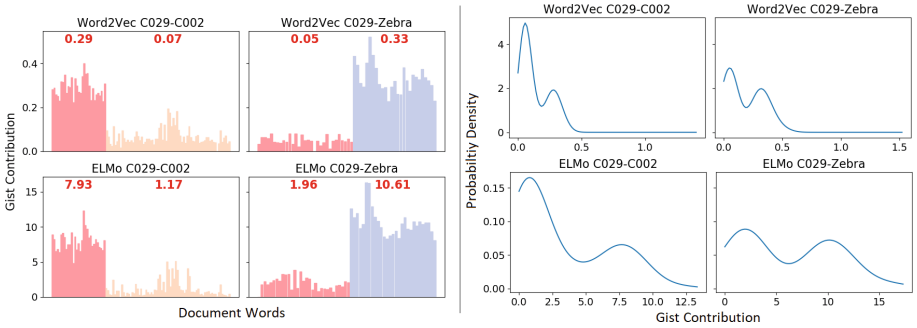

Fig. 3. Gist embedding with score (left) and corresponding KDE (right) with segment size of 50.

Figure 3 displays the effects on the gist embedding and KDE plots with a segmentation size of 50. The relationship between the gist score between highly related classes C029 and C002 becomes less distinguishable in the sense that the difference in the gist score becomes large. However, the KDEs unveil an underlying semantic difference in the distributions. In both KDEs, we see the possible existence of two classes displayed by the unimodal distributions but in both cases C029-C002 there is a higher degree of overlap between the documents displayed by the unevenness of the modes.

4 Gist Score Similarity for N-Shot Document Classification

Determining the semantic similarity between documents via N-shot learning involves solving multiple instances of a decision task given a *support set*, a small set of labelled documents $S = \{S_{C_1}, S_{C_2}, ..., S_{C_i}\}$ where S_{C_i} represents the labelled documents for class i and $|S_{c_i}| = N$ and a *query set*, $Q = \{q_1, q_2, ..., q_n\}$ of n unlabelled document to predict which is equivalent to a test set. We can define a similarity decision instance in the style of a N-shot learning instance or episode as follows:

Similarity Decision Instance: Given a single query document from the query set and a set of support documents, the goal is to match the query document to the most related support document based on some semantic aspect e.g. meaning or topic.

Based on this definition, Algorithms 2 and 3 outline the procedure to perform N-shot document classification using the gist score. Fundamentally, document classification involves determining the semantic similarity between documents which has been argued to fundamentally be a cognitive modelling problem [14]. Therefore, to some extent computational semantic similarity measures should align to some aspect of human judgement which we aim to do by aligning our method towards principals in FTT. As such, we explore the use of two types of gist known as the local and global gist to compute the gist score. It has been proposed that the gist can be differentiated into two types, the global gist which captures the meaning of an entire event as a whole and the local gist which captures the meaning of a more discrete event [17]. To some extent determining the semantic similarity between documents in a few-shot learning instance should involve both the global and local gist, In Algorithm 3 we employ the local gist to capture the discrete pairwise gist between the query and support documents whereas the global gist captures the query document in relation to the entire support set.

Algorithm 2. Gist Score Extraction, GSE($d_1, d_2, ..., d_n, size$)

Input: Documents set $d_{i:n}$, Segmentation size
Output: Gist score for each input document $d_{i:n}$

$text \leftarrow concatenate(d_1, d_2, ..., d_n)$ ▷ join all text together
$gistEmbed \leftarrow gistEmbedding(text, size)$
for i in 1 to n **do**
    $score_{d_i} \leftarrow average(gistEmbed[d_i])$ ▷ segment in embedding related to text d_i
return $score_{d_1}, score_{d_2}, ..., score_{d_n}$

Algorithm 3. Gist Score Similarity N-Shot Instance, GSS($q, S_{C_1}, S_{C_2}, ..., S_{C_n}, size$)

Input: Query document q, Support documents $S_{C_{1:n}}$, Segement size
Output: Probability of q belonging to each class $C_{1:n}$

$global \leftarrow GSE(q, S_{C_1}, S_{C_2}, ..., S_{C_n}, size)$ ▷ consider gist across all support texts
for i in 1 to n **do**
    for *document* in S_{C_i} **do** ▷ extract pairwise gist similarity
        $local_{q,S_{C_i}} \leftarrow GSE(q, document, size)$
    $gist_{q,S_{C_i}} \leftarrow average(local_{q,S_{C_i}}, global_{q,S_{C_i}})$
return $softmax(-gistq, S_{C_1}, -gistq, S_{C_1}, ..., -gistq, S_{C_n})$ ▷ probability of classes

5 Experiments

To evaluate our approach to gist extraction and reasoning we perform a series of experiments to analyse the inferencing capabilities of the gist score and baseline algorithms for few-shot learning on large documents, identify the importance of the local and global gists towards inferencing and the effect segments size in our framework has on uncovering the underlying semantic distribution.

5.1 Datasets

Popular benchmark datasets for document classification typically consist of a large number of documents with a low average number of words per document. For our experiments, we use a combination of documents collections typically used as benchmark datasets for topic segmentation[1]. These documents on average have a large number of words but also a mix of small and large documents as shown in Table 1.

Table 1. Statistics for datasets.

	Documents	Minimum words	Maximum words	Average words	Total words
Wiki animal	214	162	15696	3885	831480
Clinical	227	353	14750	2648	601171
Fiction	82	4578	53494	33542	2750472
Physics	33	6325	7956	7155	236127

[1] Dataset available at https://github.com/JaronMar/Large-Document-Dataset.

5.2 Baseline Algorithms

For baseline algorithms, we compare both few-shot document classification, distance-based classifiers and traditional document classification algorithms trained only on the N examples of each class. To compare the few-shot learning baseline algorithms we perform N-way N-shot learning with minimal pretraining to evaluate how existing models perform when access to labelled data is restrictive.

WCD: Word Centroid Distances using ELMo embedding which is an approximation of word movers distance (WMD) [13]. We apply WCD as the restrictive time complexity of n^3 to calculate the full WMD is infeasible for large documents.

Centroid: Applies a traditional but effective semantic distance measure using the euclidean distance between ELMo document centroids.

SVM: Support vector machine with linear kernel [11] using term frequency (TF-IDF) features. The penalty parameters were optimised for using grid search.

HAN: Hierarchical Attention Network trained using the same ELMo embeddings as RD_{WCD} [24]. Due to memory constraints, we train 150000 features using 50 sentences per document, 20 words per sentence over 20 epochs.

Proto: Prototypical network is a few-shot method that meta-learns to minimizes the squared Euclidean distance between the centroid of each class to its training examples in a metric/embedding space [23].

MAML: Model-Agnostic Meta-Learning is a few-shot method that meta-learns a prior over model parameters which allows the model to quickly adapt to unseen classes [8] where we set the number of inner steps to 5.

6 Results

We perform two sets of experiments; one performs binary classification on the animal and clinical datasets and the other multiclass classification using all datasets. The results show the average using the same randomly seeded support documents.

Tables 2 and 3 display the results for 1-shot classification with segment size of 50 using GSS. The results show that our proposed gist representation retains the full expressiveness of the centroid classifier and even slightly improves on the accuracy while heavily reducing the number of features. This result suggests we can accurately extract and represent the gist in large text documents. Interestingly, although taking different approaches to judge the semantic distance our result aligns closely to that of the centroid classifier. We also see both SVM and HAN, traditional document classification algorithms are not suitable for one-shot classification and report the two worst results in both cases. As for the few-shot algorithms we see that with minimal pretraining both classifiers are out performed by the unsupervised distance-based classifiers.

Table 2. Binary 1-shot classification results.

	Accuracy	Precision	Recall	F_1
WCD	0.8238	0.7521	0.9811	0.8515
Centroid	0.9807	**0.9890**	0.9724	0.9784
SVM	0.6696	0.6420	0.8097	0.7162
HAN	0.7791	0.8325	0.7796	0.7392
Proto	0.8503	0.7366	**0.9978**	0.8475
MAML	0.4852	0.5351	0.4852	0.4852
GSS_{ELMo}	**0.9814**	0.9709	0.993	**0.9813**
GSS_{W2V}	0.9093	0.8974	0.9322	0.9094
GSS_{GloVe}	0.5961	0.5754	0.7519	0.6434
GSS_{FT}	0.7429	0.7229	0.8	0.7555

Table 3. Multiclass 1-shot classification results.

	Accuracy	Precision	Recall	F_1
WCD	0.8531	0.9041	0.8678	0.8703
Centroid	0.9552	**0.9274**	**0.9718**	**0.9430**
SVM	0.5631	0.6864	0.7200	0.6398
HAN	0.7538,	0.7803	0.7842	0.7295
Proto	0.7803	0.7069	0.8554	0.6938
MAML	0.4010	0.2249	0.2830	0.1987
GSS_{ELMo}	**0.9573**	0.9251	0.9705	0.9406
GSS_{W2V}	0.9385	0.9163	0.9503	0.9258
GSS_{GloVe}	0.9160	0.8998	0.9346	0.9074
GSS_{FT}	0.8950	0.8792	0.9198	0.8819

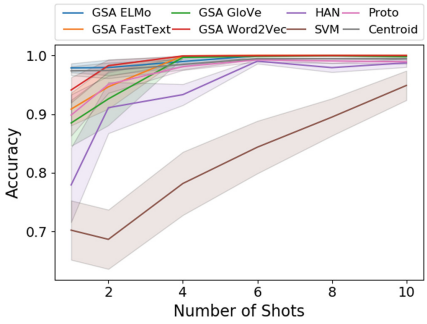

Fig. 4. Few-shot binary accuracy.

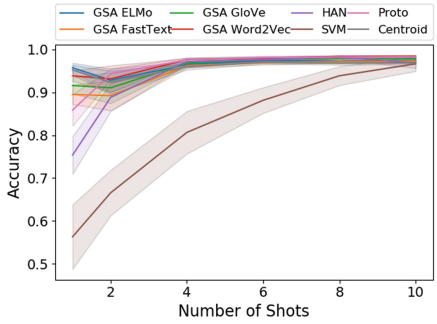

Fig. 5. Few-shot multiclass accuracy.

When increasing the number of shots (N) from 1 to 10 in Fig. 4 and 5 we expect all classifiers to generally improve. This statement holds true especially for the traditional SVM model that slowly increases the performance with more training data. In general, across all experiments GSS using ELMo performs consistently better compared to GSS with other embedding types suggesting the ELMo embeddings are semantically more meaningful. Our result also suggests for large documents we can achieve high levels of accuracy using just a single example of each class using unsupervised methods in NLP.

6.1 Effects of Segmentation

Section 3.4 shows that quantifying the gist over segment centroids can unveil an underlying semantic distribution. Figure 6 and 7 display the effects segment size has on capturing this underlying distribution based on different embedding types. The results show that segment size can significantly improve that initially under-performing GloVe and FastText models to be more comparable but still not better than ELMo and Word2Vec based embeddings. Interestingly, this

aligns with psychological experiments where fuzzier gist-based representations are better from making decisions rather than verbatim representations [1]. In both cases, we see that ELMo embeddings are robust in the sense that the accuracy is not greatly affected when the segment size is changed one again. More significantly, when the segment size is one we still get relatively high accuracy for both ELMo and Word2Vec embeddings. This implies that theses embeddings are semantically more meaningful and it is possible to represent each word embedding as a single real number and still achieve accurate results in large document classification. This is a significant implication as in general nearly all the features in the original embeddings are lost in creating the gist embedding however our results show that we can still capture the semantic similarity between documents. This suggests for certain tasks we can feasible train lower dimensional embeddings thus reducing training time and the memory requirements associated with handling large sets of embeddings.

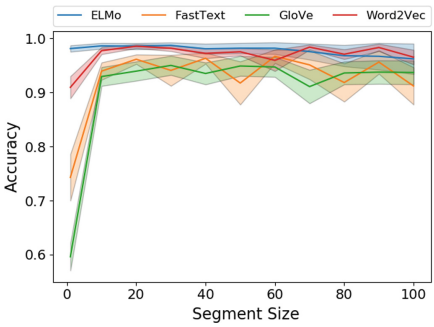

Fig. 6. Segmentation on the binary case.

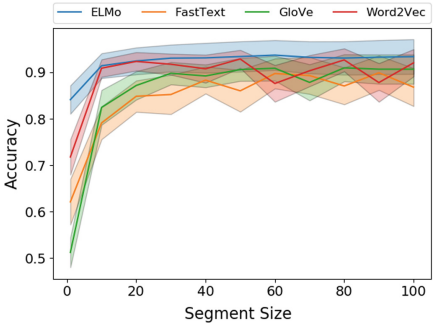

Fig. 7. Segmentation on the multiclass case.

6.2 Importance of Local and Global Gists

In Sect. 4 we introduced the concept of the global and local gists and provided a method to extract both of these gists to create the gist score for a given few-shot classification instance. The results in Fig. 8 and 9 show that generally the local gist is more effective to perform inferencing on over the global gist in both binary and multiclass classification. This makes sense as the local gist captures the discrete differences between two documents which is a more direct measure for classification opposed to the global gist. When naively combining the local and global gists by averaging, in binary classification the combined gist performs within 2–4% better or worse than the local gist whereas in the multiclass case the combined gist performs 2–20% better in all cases. Once again, intuition tells us that when there are more classes to compare the global or contextual comparison becomes more important as reflected in the results. This suggests that as the number of classes or size of the support set increases evaluating the pairwise

semantic similarity is not sufficient and global or contextual information about other classes can improve accuracy. From a psychological standpoint based on the fuzzy-preference theory, in general the combined gist is the preferred gist representation along the fuzzy-to-verbatim continua which although isn't that fuzziest representation provides still provide the most information needed for inferencing.

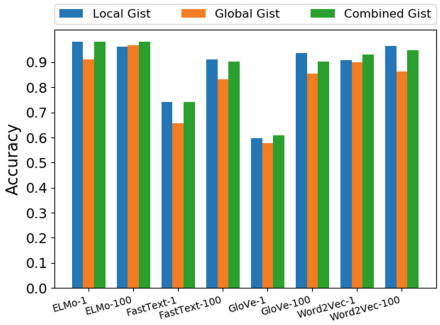

Fig. 8. Gist types on the 1-shot binary case.

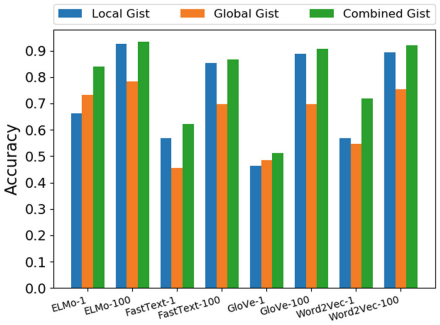

Fig. 9. Gist types on the 1-shot multi-class case.

7 Conclusion and Future Work

In this paper, we critically reduce the size of word embeddings from their original size of 300–1000 dimensions each to one dimension to create a gist representation based on psychological notions that successfully encapsulates the essential semantic information for few-shot large document classification. As future work, in this paper we assumed a sequential memory model in which the gist representation is created from the verbatim representation, but it is theorised that the gist and verbatim representations are created in parallel in FTT. It would be interesting to explore a model which learns both the verbatim and gist represents of words simultaneously as our work shows that one-dimensional gist representations for words are sufficient for large document classification and can be created from higher dimensional word embeddings, such a representation would be beneficial for memory constrained environments.

References

1. Abadie, M., Waroquier, L., Terrier, P.: Gist memory in the unconscious-thought effect. Psychol. Sci. **24**(7), 1253–1259 (2013)
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
3. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Trans. Assoc. Comput. Linguist. **5**, 135–146 (2017)

4. Brainerd, C.J., Reyna, V.F.: Gist is the grist: fuzzy-trace theory and the new intuitionism. Dev. Rev. **10**(1), 3–47 (1990)
5. Brainerd, C.J., Reyna, V.F.: Fuzzy-trace theory: Dual processes in memory, reasoning, and cognitive neuroscience (2001)
6. Douze, M., Jégou, H., Sandhawalia, H., Amsaleg, L., Schmid, C.: Evaluation of gist descriptors for web-scale image search. In: Proceedings of the ACM International Conference on Image and Video Retrieval, pp. 1–8 (2009)
7. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE Trans. Pattern Anal. Mach. Intell. **28**(4), 594–611 (2006)
8. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. CoRR abs/1703.03400 (2017). http://arxiv.org/abs/1703.03400
9. Griffiths, T.L., Steyvers, M., Tenenbaum, J.B.: Topics in semantic representation. Psychol. Rev. **114**(2), 211 (2007)
10. Hu, S., Leung, C.W., Leung, H.F., Liu, J.: To be big picture thinker or detail-oriented?: utilizing perceived gist information to achieve efficient convention emergence with bilateralism and multilateralism. In: Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, pp. 2021–2023 (2019)
11. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 137–142. Springer, Heidelberg (1998). https://doi.org/10.1007/BFb0026683
12. Jones, K.S.: A statistical interpretation of term specificity and its application in retrieval. J. Doc. **28**, 11–21 (1972)
13. Kusner, M., Sun, Y., Kolkin, N., Weinberger, K.: From word embeddings to document distances. In: International Conference on Machine Learning, pp. 957–966 (2015)
14. Lee, M.D., Pincombe, B., Welsh, M.: A comparison of machine measures of text document similarity with human judgments. In: 27th Annual Meeting of the Cognitive Science Society (CogSci 2005), pp. 1254–1259 (2005)
15. Michael Lampinen, J., Leding, J.K., Reed, K.B., Odegard, T.N.: Global gist extraction in children and adults. Memory **14**(8), 952–964 (2006)
16. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
17. Neuschatz, J.S., Lampinen, J.M., Preston, E.L., Hawkins, E.R., Toglia, M.P.: The effect of memory schemata on memory and the phenomenological experience of naturalistic situations. Appl. Cogn. Psychol. Offic. J. Soc. Appl. Res. Memory Cogn. **16**(6), 687–708 (2002)
18. Pardo, T.A.S., Rino, L.H.M., Nunes, M.G.V.: GistSumm: a summarization tool based on a new extractive method. In: Mamede, N.J., Trancoso, I., Baptista, J., das Graças Volpe Nunes, M. (eds.) PROPOR 2003. LNCS (LNAI), vol. 2721, pp. 210–218. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-45011-4_34
19. Pennington, J., Socher, R., Manning, C.: Glove: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)
20. Peters, M.E., et al.: Deep contextualized word representations. arXiv preprint arXiv:1802.05365 (2018)
21. Raunak, V.: Simple and effective dimensionality reduction for word embeddings. arXiv preprint arXiv:1708.03629 (2017)

22. Rino, L.H.M., Scott, D.: A discourse model for gist preservation. In: Borges, D.L., Kaestner, C.A.A. (eds.) SBIA 1996. LNCS, vol. 1159, pp. 131–140. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-61859-7_14
23. Snell, J., Swersky, K., Zemel, R.S.: Prototypical networks for few-shot learning. CoRR abs/1703.05175 (2017). http://arxiv.org/abs/1703.05175
24. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E.: Hierarchical attention networks for document classification. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1480–1489 (2016)
25. Zipf, G.K.: Selected studies of the principle of relative frequency in language (1932)