Revisiting Document Representations for Large-Scale Zero-Shot Learning

Jihyung Kil
The Ohio State University
Columbus, Ohio, USA
kil.5@osu.edu

Wei-Lun Chao
The Ohio State University
Columbus, Ohio, USA
chao.209@osu.edu

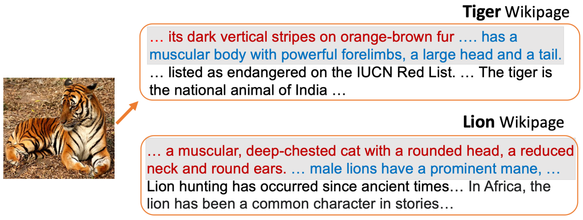
Abstract

Zero-shot learning aims to recognize unseen objects using their semantic representations. Most existing works use visual attributes labeled by humans, not suitable for large-scale applications. In this paper, we revisit the use of documents as semantic representations. We argue that documents like Wikipedia pages contain rich visual information, which however can easily be buried by the vast amount of non-visual sentences. To address this issue, we propose a semi-automatic mechanism for visual sentence extraction that leverages the document section headers and the clustering structure of visual sentences. The extracted visual sentences, after a novel weighting scheme to distinguish similar classes, essentially form semantic representations like visual attributes but need much less human effort. On the ImageNet dataset with over 10,000 unseen classes, our representations lead to a 64% relative improvement against the commonly used ones.

1 Introduction

Algorithms for visual recognition usually require hundreds of labeled images to learn how to classify an object (He et al., 2016). In reality, however, the frequency of observing an object follows a long-tailed distribution (Zhu et al., 2014): many objects do not appear frequently enough for us to collect sufficient images. Zero-shot learning (ZSL) (Lampert et al., 2009), which aims to build classifiers for unseen object classes using their *semantic representations*, has thus emerged as a promising paradigm for recognizing a large number of classes.

Being the only information of unseen objects, how well the semantic representations describe the visual appearances plays a crucial role in ZSL. One popular choice is *visual attributes* (Lampert et al., 2009; Patterson and Hays, 2012; Wah et al., 2011) carefully annotated by humans. For example, the bird "Red bellied Woodpecker" has the "capped head pattern" and "pointed wing shape". While strictly tied to visual appearances, visual attributes are laborious to collect, limiting their applicability to small-scale problems with hundreds of classes.



Tiger Wikipage

... its dark vertical stripes on orange-brown fur .... has a muscular body with powerful forelimbs, a large head and a tail. ... listed as endangered on the IUCN Red List. ... The tiger is the national animal of India ...

Lion Wikipage

... a muscular, deep-chested cat with a rounded head, a reduced neck and round ears. ... male lions have a prominent mane, ... Lion hunting has occurred since ancient times... In Africa, the lion has been a common character in stories...

Figure 1: An illustration of our ZSL approach, which recognizes the input image by comparing it to the visual sentences of documents. Here we show two documents, one for "Tiger" and one for "Lion". The gray area highlights the extracted visual sentences (red: by section headers; blue: by clustering).

For large-scale problems like ImageNet (Deng et al., 2009) that has more than $20,000$ classes, existing ZSL algorithms (Frome et al., 2013; Norouzi et al., 2013) mostly resort to *word vectors* of classes names (Mikolov et al., 2013; Pennington et al., 2014) that are automatically extracted from large corpora like Common Crawl. While almost labor free, word vectors are purely text-driven and barely aligned with visual information. As a result, the state-of-the-art ZSL accuracy on ImageNet falls far behind being practical (Changpinyo et al., 2020).

Is it possible to develop semantic representations that are as powerful as visual attributes without significant human effort? A feasibility study by representing a class with its Wikipedia page shows some positive signs — Wikipedia pages do capture rich attribute information. For example, the page "Red-bellied Woodpecker" contains phrases "red cap going from the bill to the nape" and "black and white barred patterns on their back, wings and tail" that exactly match the visual attributes mentioned above. In other words, if we can identify *visual* sentences from a document to represent a class, we are likely to attain much higher ZSL accuracy[1].

[1] Representing a class by a document has been studied in (Zhu et al., 2018; Elhoseiny et al., 2013; Qiao et al., 2016), but they use all sentences instead of extracting the visual ones.

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 3117–3128
June 6–11, 2021. ©2021 Association for Computational Linguistics

To this end, we present a simple yet effective semi-automatic approach for *visual sentence extraction*, which leverages two informative semantic cues. First, we leverage the *section structures* of Wikipedia pages: the section header indicates what kind of sentences (visual or not) appear in the section. Concretely, we search Wikipedia pages of common objects following the sysnsets in ImageNet (e.g., fish, room), and manually identify sections that contain visual information (e.g., characteristics, appearance). We then apply these visual headers to the Wikipedia pages of the remaining ImageNet classes. Second, we observe that visual sentences share some common contextual patterns: for example, they contain commonly used words or phrases of visual attributes (e.g., red color, furry surface). To leverage these patterns, we perform K-means sentence clustering using the BERT features (Devlin et al., 2018) and manually select clusters that contain visual information. We keep sentences in these clusters and combine them with those selected by section headers to represent a document. See Figure 1 for an illustration.

To further increase the discriminative ability of the visual sentences between similar object classes (e.g., breeds of dogs), we introduce a novel scheme to assign weights to sentences, emphasizing those that are more representative for each class.

We validate our approach on three datasets: ImageNet Fall 2011 dataset (Deng et al., 2009), which contains $14,840$ unseen classes with Wikipedia pages; Animals with Attributes 2 (AwA2) (Xian et al., 2018a), which has 50 animal classes; Attribute Pascal and Yahoo (aPY) (Farhadi et al., 2009), which has 32 classes. Our results are promising: compared to word vectors on ImageNet, we improve by 64% using visual sentences. On AwA2 and aPY, compared to visual attributes annotated by humans, we improve by 8% and 5%, respectively. Moreover, our new semantic representations can be easily incorporated into any ZSL algorithms. Our code and data will be available at `https://github.com/heendung/vs-zsl`.

2 Related Work

Semantic representations. Visual attributes are the most popular semantic representations (Lampert et al., 2009; Patterson and Hays, 2012; Wah et al., 2011; Zhao et al., 2019). However, due to the need of human annotation, the largest dataset has only 717 classes. Reed et al. (2016b,a) collect visual sentences for each image, which is not scalable. For large-scale recognition, word vectors (Mikolov et al., 2013) have been widely used. Lu (2015); Kampffmeyer et al. (2019); Wang et al. (2018) explore the use of WordNet hierarchy (Miller, 1995), which may not be available in other applications.

Similar to ours, Akata et al. (2015b); Elhoseiny et al. (2013); Qiao et al. (2016); Zhu et al. (2018) represent classes by documents, by counting word frequencies but not extracting visual sentences. Al-Halah and Stiefelhagen (2017) extract *single* word attributes, which are not discriminative enough (e.g., "red cap" becomes "red", "cap"). None of them works on ZSL with over 1,000 classes.

Hessel et al. (2018); Le Cacheux et al. (2020) collect images and tags of a class and derives its semantic representation from tags, which is not feasible for unseen classes on ZSL.

Zero-shot learning algorithms. The most popular way is to learn an embedding space in which visual features and semantic representations are aligned and nearest neighbor classifiers can be applied (Changpinyo et al., 2017; Romera-Paredes and Torr, 2015; Akata et al., 2015a; Kodirov et al., 2017; Schonfeld et al., 2019; Zhu et al., 2019; Xie et al., 2019; Socher et al., 2013). These algorithms consistently improve accuracy on datasets with attributes. Their accuracy on ImageNet, however, is saturated, mainly due to the poor quality of semantic representations (Changpinyo et al., 2020).

3 *Visual* Sentence Extraction

3.1 Background and notation

ZSL algorithms learn to align visual features and semantic representations using a set of *seen* classes $\mathcal{S}$. The alignment is then applied to the test images of unseen classes $\mathcal{U}$. We denote by $\mathcal{D} = \{(\boldsymbol{x}_n, y_n \in \mathcal{S})\}_{n=1}^{N}$ the training data (i.e., image feature and label pairs) with the labels coming from $\mathcal{S}$.

Suppose that we have access to a semantic representation $\boldsymbol{a}_c$ (e.g., word vectors) for each class $c \in \mathcal{S} \cup \mathcal{U}$, one popular algorithm DeViSE (Frome et al., 2013) proposes the learning objective

$$\sum_{n} \sum_{c \neq y_n} \max\{0, \Delta - f_{\boldsymbol{\theta}}^{\top}(\boldsymbol{x}_n) \boldsymbol{M} g_{\boldsymbol{\phi}}(\boldsymbol{a}_{y_n}) + f_{\boldsymbol{\theta}}^{\top}(\boldsymbol{x}_n) \boldsymbol{M} g_{\boldsymbol{\phi}}(\boldsymbol{a}_c)\}, \quad (1)$$

where $\Delta \geq 0$ is a margin. That is, DeViSE tries to learn transformations $f_{\boldsymbol{\theta}}$ and $g_{\boldsymbol{\phi}}$ and a matrix $\boldsymbol{M}$ to maximize the visual and semantic alignment of

Section headers
Characteristics, Description, Appearance, Habitat, Diet, Construction and Mechanics, Materials for utensil, Design for appliance, Furnishings for room, Fabrication, Feature for geological formation, Design, Equipment for sport
History, Health, Terminology, Mythology, Conservation, Culture, References, External links, Further reading

Table 1: Visual (top) & Non-Visual (bottom) sections.

Sentence clusters
It has large ears that help the fox lower its body temperature. It usually has a gray coat, with rusty tones, and a black tip to its tail. It has distinct dark patches around the nose. It is most recognisable for its dark vertical stripes on orangish-brown fur. $\cdots$ muscular body with powerful forelimbs, a large head and a tail. They have a mane-like heavy growth of fur around the neck and jaws $\cdots$
The kit fox is a socially monogamous species. Male and female kit foxes usually establish monogamous mating $\cdots$ The average lifespan of a wild kit fox is 5.5 years. Tiger mates all year round, but most cubs are born between March $\cdots$ The father generally takes no part in rearing. The mortality rate of tiger cubs is about 50% in the first two years.

Table 2: Sentence clusters. The top cluster is *visual* and the bottom one is *non-visual*. The sentences from a class *kit-fox* are in red and those from a class *tiger* are in blue.

the same classes while minimizing that between classes. We can then classify a test image $\boldsymbol{x}$ by

$$\arg\max_{c\in\mathcal{U}} f_{\boldsymbol{\theta}}^{\top}(\boldsymbol{x})\boldsymbol{M}g_{\boldsymbol{\phi}}(\boldsymbol{a}_c). \tag{2}$$

Here, we consider that every class $c \in \mathcal{S}\cup\mathcal{U}$ is provided with a document $H_c = \{\boldsymbol{h}_1^{(c)}, \cdots, \boldsymbol{h}_{|H_c|}^{(c)}\}$ rather than $\boldsymbol{a}_c$, where $|H_c|$ is the amount of sentences in document H_c and $\boldsymbol{h}_j^{(c)}$ is the jth sentence, encoded by BERT (Devlin et al., 2018). We mainly study DeViSE, but our approach can easily be applied to other ZSL algorithms.

3.2 Visual section selection

We aim to filter out sentences in H_c that are not describing visual information. We first leverage the section headers in Wikipedia pages, which indicate what types of sentences (visual or not) are in the sections. For example, the page "Lion" has sections "Description" and "Colour variation" that are likely for visual information, and "Health" and "Cultural significance" that are for non-visual information.

To efficiently identify these section headers, we use ImageNet synsets (Deng et al., 2009), which group objects into 16 broad categories. We randomly sample $30 \sim 35$ classes per group, resulting in a set of 500 classes. We then retrieve the corresponding Wikipedia pages by their names and manually identify section headers related to visual sentences. By sub-sampling classes in this way, we can quickly find section headers that are applicable to other classes within the same groups. Table 1 shows some visual/non-visual sections gathered from the 500 classes. For example, "Characteristics" frequently appears in pages of animals to describe their appearances. In contrast, sections like "History" or "Mythology" do not contain visual information. Investigating all the 500 Wikipedia pages carefully, we find 40 distinct visual sections. We also include the first paragraph of a Wikipedia page, which often contains visual information.

3.3 Visual cluster selection

Our second approach uses K-means for sentence clustering: visual sentences often share common words and phrases of visual attributes, naturally forming clusters. We represent each sentence using the BERT features (Devlin et al., 2018), and perform K-means (with $K = 100$) over all the sentences from Wikipedia pages of ImageNet classes. We then manually check the 100 clusters and identify 40 visual clusters. Table 2 shows a visual (top) and a non-visual (bottom) cluster. We highlight sentences related to two classes: "kit-fox" (red) and "tiger" (blue). The visual cluster describes the animals' general appearances, especially about visual attributes "dark", "black", "tail", "large", etc. In contrast, the non-visual cluster describes mating and lifespan that are not related to visual aspects.

3.4 Semantic representations of documents

After we obtain a filtered document $\hat{H}_c$, which contains sentences of the *visual* sections and clusters, the next step is to represent $\hat{H}_c$ by a vector $\boldsymbol{a}_c$ so that nearly all the ZSL algorithms can leverage it.

A simple way is **average**, $\bar{\boldsymbol{a}}_c = \frac{1}{|\hat{H}_c|}\sum_{\boldsymbol{h}\in\hat{H}_c}\boldsymbol{h}$, where $\boldsymbol{h}$ is the BERT feature. This, however, may not be discriminative enough to differentiate similar classes that share many common descriptions (e.g., dog classes share common phrase like "a breed of dogs" and "having a coat or a tail").

We therefore propose to identify informative sentences that can enlarge the difference of $\boldsymbol{a}_c$ between classes. Concretely, we learn to assign each sentence a weight λ, such that the resulting **weighted average** $\boldsymbol{a}_c = \frac{1}{|\hat{H}_c|}\sum_{\boldsymbol{h}\in\hat{H}_c}\lambda(\boldsymbol{h}) \times \boldsymbol{h}$ can be more distinctive. We model $\lambda(\cdot) \in \mathbb{R}$ by a multi-layer perceptron (MLP) $b_{\boldsymbol{\psi}}$

$$\lambda(\boldsymbol{h}) = \frac{\exp(b_{\boldsymbol{\psi}}(\boldsymbol{h}))}{\sum_{\boldsymbol{h}'\in\hat{H}_c}\exp(b_{\boldsymbol{\psi}}(\boldsymbol{h}'))}. \tag{3}$$

We learn $b_{\boldsymbol{\psi}}$ to meet two criteria. On the one hand, for very similar classes c and c' whose similarity $\cos(\boldsymbol{a}_c, \boldsymbol{a}_{c'})$ is larger than a threshold τ, we want

$\cos(\boldsymbol{a}_c, \boldsymbol{a}_{c'})$ to be smaller than τ so they can be discriminable. On the other hand, for other pair of less similar classes, we want their similarity to follow the **average** semantic representation $\bar{\boldsymbol{a}}_c$[2].

To this end, we initialize b_ψ such that the initial $\boldsymbol{a}_c$ is close to $\bar{\boldsymbol{a}}_c$. We do so by first learning b_ψ to minimize the following objective

$$\sum_{c\in\mathcal{S}\cup\mathcal{U}} \max\{0, \epsilon - \cos(\boldsymbol{a}_c, \bar{\boldsymbol{a}}_c)\}. \tag{4}$$

We set $\epsilon = 0.9$, forcing $\boldsymbol{a}_c$ and $\bar{\boldsymbol{a}}_c$ of the same class to have $\cos(\boldsymbol{a}_c, \bar{\boldsymbol{a}}_c) > 0.9$. We then fine-tune b_ψ by minimizing the following objective

$$\sum_{c}^{\mathcal{S}\cup\mathcal{U}} \sum_{c'\neq c}^{\mathcal{S}\cup\mathcal{U}} \max\{0, \cos(\boldsymbol{a}_c, \boldsymbol{a}_{c'}) - \tau\}. \tag{5}$$

We assign τ a high value (e.g., 0.95) to only penalize overly similar semantic representations. Please see the appendix for details.

Comparison. Our approach is different from DAN (Iyyer et al., 2015). First, we learn an MLP to assign weights to sentences so that their embeddings can be combined appropriately to differentiate classes. In contrast, DAN computes the averaged embedding and learns an MLP to map it to another (more discriminative) embedding space. Second, DAN leans the MLP with a classification loss. In contrast, we learn the MLP to reduce the embedding similarity between similar classes while maintaining the similarity for other pairs of classes.

4 Experiments

4.1 Dataset and splits: ImageNet

We use the ImageNet Fall 2011 dataset (Deng et al., 2009) with $21,842$ classes. We use the 1K classes in ILSVRC 2012 (Russakovsky et al., 2015) for DeViSE training and validation (cf. Equation 1), leaving the remaining $20,842$ classes as unseen classes for testing. We follow (Changpinyo et al., 2016) to consider three tasks, 2-Hop, 3-Hop, and ALL, corresponding to **1,290**, **5,984**, and **14,840** unseen classes *that have Wikipedia pages and word vectors* and are within two, three, and arbitrary tree hop distances (w.r.t. the ImageNet hierarchy) to the 1K classes. On average, each page contains **80** sentences. For images, we use the $2,048$-dimensional ResNet visual features (He et al., 2016) provided by Xian et al. (2018a). For sentences, we use a 12-layer pre-trained BERT model (Devlin et al., 2018). We denote by BERT_p the pre-trained BERT and BERT_f the one fine-tuned with DeViSE. Please see the appendix for details.

Model	Type	Filter	2-Hop	3-Hop	ALL
Random	-	-	0.078	0.017	0.007
DeViSE	w2v-v2	-	6.45	1.99	0.78
	BERT_p	No	6.73	2.23	0.83
DeViSE*	w2v-v2	-	11.55	3.07	1.48
	BERT_p	No	13.84	4.05	1.75
		Vis_sec	15.56	4.41	1.82
		Vis_clu	15.72	4.49	2.01
		$\text{Vis}_\text{sec-clu}$	15.86	4.65	2.05
	$\text{BERT}_\text{p-w}$	$\text{Vis}_\text{sec-clu}$	16.32	4.73	2.10
	BERT_f	No	17.70	5.17	2.29
		Vis_sec	19.52	5.20	2.32
		Vis_clu	19.74	5.37	2.36
		$\text{Vis}_\text{sec-clu}$	19.82	5.39	2.39
	$\text{BERT}_\text{f-w}$	$\text{Vis}_\text{sec-clu}$	20.47	5.53	2.42
EXEM	w2v-v2	-	16.04	4.54	1.99
	BERT_f	$\text{Vis}_\text{sec-clu}$	21.22	5.42	2.37
HVE	w2v-v2	-	8.63	2.38	1.09
	$\text{BERT}_\text{f-w}$	$\text{Vis}_\text{sec-clu}$	18.42	5.12	2.07

Table 3: Comparison of different semantic representations on ImageNet. We use *per-class* Top-1 accuracy(%). The best is in red and the second best in blue.

4.2 Baselines, variants, and metrics

Word vectors of class names are the standard semantic representations for ImageNet. Here we compare to the state-of-the-art **w2v-v2** provided by Changpinyo et al. (2020), corresponding to a skip-gram model (Mikolov et al., 2013) trained with ten passes of the Wikipedia dump corpus. For ours, we compare using all sentences **(NO)**, visual sections **($\textbf{Vis}_\textbf{sec}$)** or visual clusters **($\textbf{Vis}_\textbf{clu}$)**, and both **($\textbf{Vis}_\textbf{sec-clu}$)**. On average, $\textbf{Vis}_\textbf{sec-clu}$ filters out **57**% of the sentences per class. We denote **weighted average** (Section 3.4) by $\text{BERT}_\text{p-w}$ and $\text{BERT}_\text{f-w}$.

The original DeViSE (Frome et al., 2013) has $f_{\boldsymbol{\theta}}$ and $g_{\boldsymbol{\phi}}$ as identity functions. Here, we consider a stronger version, DeViSE*, in which we model $f_{\boldsymbol{\theta}}$ and $g_{\boldsymbol{\phi}}$ each by a two-hidden layers multi-layer perceptron (MLP). We also experiment with two state-of-the-art ZSL algorithms, EXEM (Changpinyo et al., 2020) and HVE (Liu et al., 2020).

We use the average *per-class* Top-1 classification accuracy as the metric (Xian et al., 2018a).

4.3 Main results

Table 3 summarizes the results on ImageNet. In combining with each ZSL algorithm, our semantic representations $\textbf{Vis}_\textbf{sec-clu}$ that uses visual sections

[2] The purpose of introducing $\lambda(\cdot)$ is to improve $\boldsymbol{a}_c$ from the average representation $\bar{\boldsymbol{a}}_c$ to differentiate similar classes.

Model	Type	AwA2				aPY			
		ZSL	GZSL			ZSL	GZSL		
			U	S	H		U	S	H
DeViSE	Visual attributes	59.70	17.10	74.70	27.80	37.02	3.54	78.41	6.73
	w2v-v2	39.56	2.18	69.29	4.22	27.67	1.68	85.53	3.22
	BERT_p + $\text{Vis}_\text{sec-clu}$	64.32	19.79	72.46	31.09	38.79	3.94	71.60	7.51

Table 4: Results on AwA2 and aPY. We compare different semantic representations. Visual attributes are annotated by humans. **GZSL** is the generalized ZSL setting (Xian et al., 2018a). In GZSL, **U**, **S**, **H** denote unseen class accuracy, seen class accuracy, and their harmonic mean, respectively. We use *per-class* Top-1 accuracy (%).

and visual clusters for sentence extraction outperforms **w2v-v2**. More discussions are as follows.

BERT vs. w2v-v2. For both DeViSE* and DeViSE, BERT_p by averaging all the sentences in a Wikipedia page outperforms w2v-v2, suggesting that representing a class by its document is more powerful than its word vector.

DeViSE* vs. DeViSE. Adding MLPs to DeViSE largely improves its accuracy: from 0.78% (DeViSE + w2v-v2) to 1.48% (DeViSE* + w2v-v2) at ALL. In the following, we then focus on DeViSE*.

Visual sentence extraction. Comparing different strategies for BERT_p, we see both $\textbf{Vis}_\textbf{clu}$ and $\textbf{Vis}_\textbf{sec}$ largely improves **NO**, demonstrating the effectiveness of sentence selection. Combining the two sets of sentences ($\textbf{Vis}_\textbf{sec-clu}$) leads to a further boost.

Fine-tuning BERT. BERT can be fine-tuned together with DeViSE*. The resulting BERT_f has a notable gain over BERT_p (e.g., 2.39% vs. 2.05%).

Weighted average. With the weighted average ($\text{BERT}_\text{p-w}$, $\text{BERT}_\text{f-w}$), we obtain the best accuracy.

ZSL algorithms. EXEM + w2v-v2 outperforms DeViSE* + w2v-v2, but falls behind DeViSE* + $\text{BERT}_\text{p-w}$ (or BERT_f, $\text{BERT}_\text{f-w}$). This suggests that algorithm design and semantic representations are both crucial. Importantly, EXEM and HVE can be improved using our proposed semantic representations, demonstrating the applicability and generalizability of our approach.

4.4 Results on other datasets

Table 4 summarizes the results on AwA2 (Xian et al., 2018a) and aPY (Farhadi et al., 2009). The former has 40 seen and 10 unseen classes; the latter has 20 seen and 12 unseen classes. We apply DeViSE together with the 2,048-dimensional ResNet features (He et al., 2016) provided by Xian et al. (2018a). Our proposed semantic representations (i.e., $\textbf{BERT}_\textbf{p}$ + $\text{Vis}_\text{sec-clu}$) outperform **w2-v2** and the manually annotated visual attributes on both the ZSL and generalized ZSL (GZSL) settings. Please see the appendix for the detailed experimental setup. These improved results on ImageNet, AwA2, and aPY demonstrate our proposed method's applicability to multiple datasets.

Model	Type	Filter	2-Hop	3-Hop	ALL
DeViSE*	BERT_p	No	13.84	4.05	1.75
	$\text{BERT}_\text{p-w-direct}$	No	14.85	4.25	1.79
	BERT_p	Par_1st	13.48	4.10	1.78
		Cls_name	14.82	3.31	1.40
		Vis_sec	15.56	4.41	1.82
		Vis_clu	15.72	4.49	2.01
		$\text{Vis}_\text{sec-clu}$	15.86	4.65	2.05
	$\text{BERT}_\text{p-w}$	$\text{Vis}_\text{sec-clu}$	16.32	4.73	2.10

Table 5: The effectiveness of our visual sentence extraction. $\textbf{BERT}_\textbf{p-w-direct}$ directly learns visual sentences without our sentence selection. $\textbf{Par}_\textbf{1st}$ and $\textbf{Cls}_\textbf{name}$ use the first paragraph and sentences containing the class name, respectively.

4.5 Analysis on ImageNet

To further justify the effectiveness of our approach, we compare to additional baselines in Table 5.

- $\textbf{BERT}_\textbf{p-w-direct}$: it directly learns b_ψ (Equation 3) as part of the DeViSE objective. Namely, we directly learn b_ψ to identify visual sentences, without our proposed selection mechanisms, such that the resulting $\boldsymbol{a}_c$ optimizes Equation 1.
- $\textbf{Par}_\textbf{1st}$: it uses the first paragraph of a document.
- $\textbf{Cls}_\textbf{name}$: it uses the sentences of a Wikipedia page that contain the class name.

As shown in Table 5, our proposed sentence selection mechanisms (i.e., Vis_sec, Vis_clu, and $\text{Vis}_\text{sec-clu}$) outperform all the three baselines.

5 Conclusion

ZSL relies heavily on the quality of semantic representations. Most recent work, however, focuses solely on algorithm design, trying to squeeze out the last bit of information from the pre-define, likely poor semantic representations. Changpinyo et al. (2020) has shown that existing algorithms are trapped in the plateau of inferior semantic representations. Improving the representations is thus more crucial for ZSL. We investigate this direction and show promising results by extracting *distinctive visual* sentences from documents for representations, which can be easily used by any ZSL algorithms.

Acknowledgment

This research is supported by the OSU GI Development funds. We are thankful for the support of computational resources by Ohio Supercomputer Center and AWS Cloud Credits for Research.

References

Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. 2015a. Label-embedding for image classification. *TPAMI*, 38(7):1425–1438.

Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. 2015b. Evaluation of output embeddings for fine-grained image classification. In *CVPR*.

Ziad Al-Halah and Rainer Stiefelhagen. 2017. Automatic discovery, association estimation and learning of semantic attributes for a thousand categories. In *CVPR*.

Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. 2016. Synthesized classifiers for zero-shot learning. In *CVPR*.

Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. 2020. Classifier and exemplar synthesis for zero-shot learning. *IJCV*, 128(1):166–201.

Soravit Changpinyo, Wei-Lun Chao, and Fei Sha. 2017. Predicting visual exemplars of unseen classes for zero-shot learning. In *ICCV*.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *CVPR*. Ieee.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Mohamed Elhoseiny, Babak Saleh, and Ahmed Elgammal. 2013. Write a classifier: Zero-shot learning using purely textual descriptions. In *ICCV*.

Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. 2009. Describing objects by their attributes. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1778–1785. IEEE.

Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. 2013. Devise: A deep visual-semantic embedding model. In *NIPS*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*.

Jack Hessel, David Mimno, and Lillian Lee. 2018. Quantifying the visual concreteness of words and topics in multimodal datasets. *arXiv preprint arXiv:1804.06786*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. *arXiv preprint arXiv:1902.00751*.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *ACL*.

Michael Kampffmeyer, Yinbo Chen, Xiaodan Liang, Hao Wang, Yujia Zhang, and Eric P Xing. 2019. Rethinking knowledge graph propagation for zero-shot learning. In *CVPR*.

Elyor Kodirov, Tao Xiang, and Shaogang Gong. 2017. Semantic autoencoder for zero-shot learning. In *CVPR*.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of bert. *arXiv preprint arXiv:1908.08593*.

Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. 2009. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*.

Yannick Le Cacheux, Adrian Popescu, and Herve Le Borgne. 2020. Webly supervised semantic embeddings for large scale zero-shot learning. In *Proceedings of the Asian Conference on Computer Vision*.

Shaoteng Liu, Jingjing Chen, Liangming Pan, Chong-Wah Ngo, Tat-Seng Chua, and Yu-Gang Jiang. 2020. Hyperbolic visual embedding learning for zero-shot recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9273–9281.

Yao Lu. 2015. Unsupervised learning on neural network outputs: with application in zero-shot learning. *arXiv preprint arXiv:1506.00990*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. 2013. Zero-shot learning by convex combination of semantic embeddings. *arXiv preprint arXiv:1312.5650*.

Genevieve Patterson and James Hays. 2012. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *CVPR*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Ruizhi Qiao, Lingqiao Liu, Chunhua Shen, and Anton Van Den Hengel. 2016. Less is more: zero-shot learning from online textual documents with noise suppression. In *CVPR*.

Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. 2016a. Learning deep representations of fine-grained visual descriptions. In *CVPR*.

Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. 2016b. Generative adversarial text to image synthesis. In *ICML*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Bernardino Romera-Paredes and Philip Torr. 2015. An embarrassingly simple approach to zero-shot learning. In *ICML*.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252.

Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. 2019. Generalized zero-and few-shot learning via aligned variational autoencoders. In *CVPR*.

Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. 2013. Zero-shot learning through cross-modal transfer. In *Advances in neural information processing systems*, pages 935–943.

Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2008. Yago: A large ontology from wikipedia and wordnet. *Journal of Web Semantics*, 6(3):203–217.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. 2011. The caltech-ucsd birds-200-2011 dataset.

Xiaolong Wang, Yufei Ye, and Abhinav Gupta. 2018. Zero-shot recognition via semantic embeddings and knowledge graphs. In *CVPR*.

Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. 2018a. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *TPAMI*, 41(9):2251–2265.

Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. 2018b. Feature generating networks for zero-shot learning. In *CVPR*.

Yongqin Xian, Saurabh Sharma, Bernt Schiele, and Zeynep Akata. 2019. f-vaegan-d2: A feature generating framework for any-shot learning. In *CVPR*.

Guo-Sen Xie, Li Liu, Xiaobo Jin, Fan Zhu, Zheng Zhang, Jie Qin, Yazhou Yao, and Ling Shao. 2019. Attentive region embedding network for zero-shot learning. In *CVPR*.

Amrapali Zaveri, Dimitris Kontokostas, Mohamed A Sherif, Lorenz Bühmann, Mohamed Morsey, Sören Auer, and Jens Lehmann. 2013. User-driven quality evaluation of dbpedia. In *Proceedings of the 9th International Conference on Semantic Systems*.

Bo Zhao, Yanwei Fu, Rui Liang, Jiahong Wu, Yonggang Wang, and Yizhou Wang. 2019. A large-scale attribute dataset for zero-shot learning. In *CVPRW*.

Pengkai Zhu, Hanxiao Wang, and Venkatesh Saligrama. 2019. Generalized zero-shot recognition based on visually semantic embedding. In *CVPR*.

Xiangxin Zhu, Dragomir Anguelov, and Deva Ramanan. 2014. Capturing long-tail distributions of object subcategories. In *CVPR*.

Yizhe Zhu, Mohamed Elhoseiny, Bingchen Liu, Xi Peng, and Ahmed Elgammal. 2018. A generative adversarial approach for zero-shot learning from noisy texts. In *CVPR*.

Appendix

In this appendix, we provide details omitted in the main text.

A Contribution

Our contribution is not merely in the method we developed, but also in the direction we explored. As discussed in Section 5 of the main paper, most of the efforts in ZSL have focused on algorithm design to associate visual features and pre-defined semantic representations. Yet, it is also important to improve semantic representations. Indeed, one reason that ZSL performs poorly on large-scale datasets is the poor semantic representations (Changpinyo et al., 2020). We therefore chose to investigate this direction by revisiting document representations, with the goal to make our contributions widely applicable. To this end, we deliberately kept our method simple and intuitive, but also provided insights for future work to build upon. Our manual inspection identified important properties of visual sentences like the clustering structure, enabling us to efficiently extract them. We chose to not design new ZSL algorithms but make our semantic representations compatible with existing ones to clearly demonstrate the effectiveness of improving semantic representations.

B More Related Work

Zero-shot learning (ZSL) algorithms construct visual classifiers based on semantic representations. Some recent work applies generative models to generate images or visual features of unseen classes (Xian et al., 2019, 2018b; Zhu et al., 2018), so that conventional supervised learning algorithms can be applied.

Knowledge bases usually contain triplets of entities and relationships. The entities are usually objects, locations, etc. For ZSL, we need entities to be fine-grained (e.g., “beaks”) and capture more visual appearances. YAGO (Suchanek et al., 2008) and DBpedia (Zaveri et al., 2013) leverage Wikipedia infoboxes to construct triplets, which is elegant but not suitable for ZSL since Wikipedia infoboxes contain insufficient visual information. Thus, these datasets and construction methods may not be directly applicable to ZSL. Nevertheless, the underlying methodologies are inspiring and could serve as the basis for future work. The datasets also offer inter-class relationships that are complementary to visual descriptions, and may be useful to establish class relationships in ZSL algorithms like SynC (Changpinyo et al., 2016).

C Statistics of Wikipedia Pages

We use a Wikipedia API to extract pages from Wikipedia for ImageNet 21,842 classes. Among 21,842 classes, we find that some classes have multiple Wikipedia pages because of their ambiguous class names. For example, a class “*black widow*” in ImageNet refers to a spider with dark brown or a shiny black in colour, but it also refers to the name of a “*Marvel Comics*” character in Wikipedia. We therefore exclude such classes and also classes that do not have word vectors, resulting in 15,833 classes. The Wikipedia pages of the 15K classes contain 1,260,889 sentences where each class has 80 sentences on average. We also investigate the number of sentences by our filters (i.e. $\textbf{Vis}_{\textbf{sec}}$, $\textbf{Vis}_{\textbf{cls}}$, $\textbf{Vis}_{\textbf{sec-clu}}$). As a result, we correspondingly find 213,585, 534,852, 542,645 sentences, which are 16%, 42%, 43% of all sentences in 15K classes, respectively (See Figure 2).

D Weighted Average Representations

D.1 Observation

Two similar classes may have similar averaged visual sentence embeddings since they share many

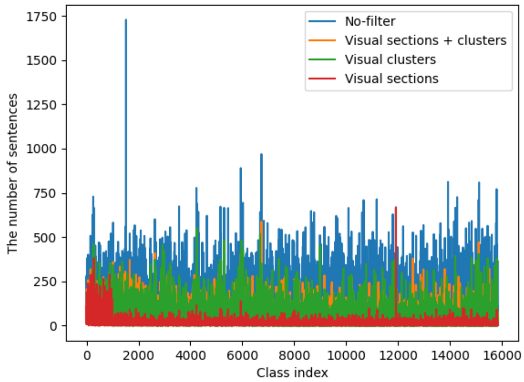


Figure 2: Statistics of **Wikipedia** pages.

common descriptions. For example, Figure 3 shows that the averaged embedding (i.e., BERT_p and BERT_f) between "Kerry Blue Terrier" and "Soft-coated Terrier" are overly similar since they share a number of sentences containing the common dog features such as "a breed of dog" or "having a coat or a tail". Thus, if we represent their semantic representations $\boldsymbol{a}_c$ as the averaged embeddings, ZSL models may not differentiate them.

D.2 Algorithm

In Section 3.4 of the main text, we introduce $\lambda(\cdot)$ to give each sentence $\boldsymbol{h}$ of a document a weight. We note that, while learning $\lambda(\cdot)$ can enlarge the distance of $\boldsymbol{a}_c$ between similar classes, we should not overly maximize the distance to prevent semantically similar classes (e.g., different breed of dogs) end up being less similar than dissimilar classes (e.g., dogs and cats). To this end, we introduce a margin loss with τ in Equation 5, which only penalize overly similar semantic representations.

We also note that, the purpose of $\lambda(\cdot)$ is to improve $\boldsymbol{a}_c$ from the simple **average** embedding $\bar{\boldsymbol{a}}_c$. We therefore initialize $\lambda(\cdot)$ such that the initial $\boldsymbol{a}_c$ is similar to $\bar{\boldsymbol{a}}_c$. We do so by first learning b_ψ with the following objective:

$$\sum_{c\in\mathcal{S}\cup\mathcal{U}} \max\{0, \epsilon - \cos(\boldsymbol{a}_c, \bar{\boldsymbol{a}}_c)\}. \qquad (6)$$

We set $\epsilon = 0.9$, forcing $\boldsymbol{a}_c$ and $\bar{\boldsymbol{a}}_c$ to have a similarity larger than 0.9.

D.3 Results

Figure 3 demonstrates the effectiveness of the weighted average embedding $\text{BERT}_\text{f-w}$. While other semantic representations predict "Kerry Blue Terrier" as other similar dog, "soft-coated Terrier", $\text{BERT}_\text{f-w}$ is able to classify the image correctly. In addition, based on the attention weights, we report the Top 3 sentences and the Bottom 3 sentences. The Top 1st sentence contains the inherent features for "Kerry Blue Terrier" such as *long head* or *soft-to-curly coat* while the Top 2nd and 3rd sentences describe general features of dogs. On the other hand, the Bottom 3 sentences do not have visual appearance of the object. This suggest that our weighted representation $\text{BERT}_\text{f-w}$ is more representative to "Kerry Blue Terrier" than other semantic representations.

E Dataset, Features, Metrics, and ZSL Algorithm

For visual features, we use the $2,048$-dimensional ResNet visual features (He et al., 2016) provided by Xian et al. (2018a). Word vectors can be found in (Changpinyo et al., 2020). Followed by (Xian et al., 2018a), we use the average *per-class* Top-1 accuracy as our metric. Instead of simply averaging over all test images (i.e. the average *per-sample* Top-1 accuracy), this accuracy is obtained by first taking average over all images in each test class independently and then taking average over all test classes. Compared to the average *per-sample* accuracy, the *per-class* accuracy is a more suitable for ImageNet since the dataset is highly imbalanced (Changpinyo et al., 2020). The state-of-the-art algorithms in ZSL are EXEM and HVE proposed by (Changpinyo et al., 2020) and (Liu et al., 2020), respectively. To make fair comparison with our models, we evaluate their algorithms on the same number of our test classes using their official codes.

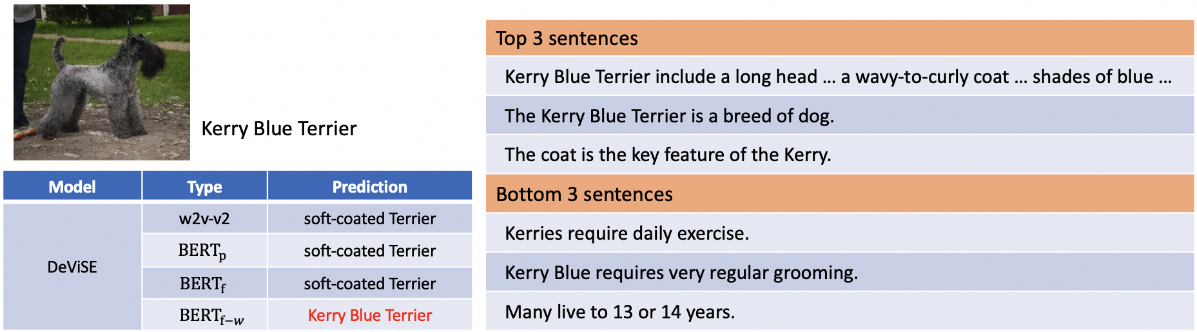
E.1 ImageNet

We follow (Xian et al., 2018a; Changpinyo et al., 2016) to consider three tasks, 2-Hop, 3-Hop, and ALL, corresponding to $1,509, 7,678$ and $20,345$ unseen classes that have word vectors and are within two, three, and arbitrary tree hop distances to the $1,000$ seen classes.

We search Wikipedia and successfully retrieve pages for **15,833** classes, of which **1,290**, **5,984**, and **14,840** are for 2-Hop, 3-Hop, and ALL.

E.2 AwA2

Animals with Attributes2 (AwA2) provides 37,322 images of 50 animal classes. On average, each class



Model	Type	Prediction
DeViSE	w2v-v2	soft-coated Terrier
	BERT$_p$	soft-coated Terrier
	BERT$_f$	soft-coated Terrier
	BERT$_{f-w}$	Kerry Blue Terrier

Top 3 sentences
Kerry Blue Terrier include a long head ... a wavy-to-curly coat ... shades of blue ...
The Kerry Blue Terrier is a breed of dog.
The coat is the key feature of the Kerry.
Bottom 3 sentences
Kerries require daily exercise.
Kerry Blue requires very regular grooming.
Many live to 13 or 14 years.

Figure 3: Qualitative analysis of a class *Kerry Blue Terrier*. w2v-v2, BERT$_\text{p}$, and BERT$_\text{f}$ can not distinguish between *Kerry Blue Terrier* and *Soft-coated Terrier* since two classes share the common features of dogs such as "a breed of dog" or "having a coat or a tail". On the other hand, our weighted average BERT$_\text{f-w}$ is able to differentiate them by weighting on the sentences. We report the Top 3 sentences and the Bottom 3 sentences based on the attention weights.

includes 746 images. It also provides 85 visual attributes that are manually annotated by humans. In AwA2, classes are split into 40 seen classes and 10 unseen classes. For GZSL, a total of 50 classes is used for testing.

E.3 aPY

Attribute Pascal and Yahoo (aPY) contains 15,339 images of 32 classes with 64 attributes. The classes are split into 20 seen classes and 12 unseen classes. A total of 32 classes is used for testing on GZSL.

E.4 DeViSE (Frome et al., 2013) vs. EXEM (Changpinyo et al., 2020) vs. HVE (Liu et al., 2020)

All algorithms learn feature transformations to associate visual features $\boldsymbol{x}$ and semantic representations $\boldsymbol{a}_c$. The key differences are what and how to learn. DeViSE* learns two MLPs $f_{\boldsymbol{\theta}}$ and $g_{\boldsymbol{\phi}}$ to embed $\boldsymbol{x}$ and $\boldsymbol{a}_c$ into a common space, while HVE embeds them into a hyperbolic space. EXEM learns kernel regressors to embed $\boldsymbol{a}_c$ into the visual space. On how to learn, DeViSE* and HVE force each image $\boldsymbol{x}$ to be similar to the true class $\boldsymbol{a}_c$ by a margin loss and a ranking loss respectively, while EXEM learns to regress the averaged visual features of a class from $\boldsymbol{a}_c$.

F Implementation Details

F.1 Sentence representations from BERT

Sentence representations can be defined in multiple ways such as a [CLS] token embbedding or an average word embedding from different layers in BERT (Reimers and Gurevych, 2019). In our experiments, the average word embedding from the second last layer of BERT achieve the best results in all cases.

Model	Type	Filter	Threshold τ	2-Hop
DeViSE*	BERT$_\text{p-w}$	Vis$_\text{sec-clu}$	0.98	15.97
			0.97	16.09
			0.96	16.32
			0.95	16.13
	BERT$_\text{f-w}$	Vis$_\text{sec-clu}$	0.88	20.34
			0.86	20.44
			0.82	20.33
			0.80	20.47

Table 6: Results of per-class Top-1 accuracy(%) on 2-Hop with different thresholds τ and semantic representation types. The best is in red and the second best in blue.

F.2 Hyperparameters

DeViSE (Frome et al., 2013) has a tunable margin $\Delta \geq 0$ (cf. Section 3.1 in the main text) which its default value is 0.1. We try multiple values 0.1, 0.2, 0.5, and 0.7 to find the best setting. DeViSE uses Adam optimizer which its learning rate is $1e^{-3}$ by default. We try different possible values, $1e^{-3}$, $5e^{-4}$, $2e^{-4}$, and $1e^{-4}$. Among all 16 possible combination of the margin and learning rate, we find that margin of $\mathbf{0.2}$ and learning rate of $\mathbf{2e^{-4}}$ achieve the best results on all our cases.

F.3 Fine-tuned models

For fine-tuning, DeViSE* is first attached to a BERT model. Then, we train the model with jointly fine-tuning BERT parameters based on the DeViSE* objective. Regards to BERT training, Houlsby et al. (2019) demonstrates that fine-tuning only last few n layers (e.g. 2 or 4) can outperform fine-tuning all layers in some NLP tasks. Kovaleva et al. (2019) also shows that the fine-tuning procedure is more effective to the last few layers than earlier layers. Considering the computational resources and time, we therefore set n equal to 2. After fine-tuning, we freeze BERT parameters and further train DeViSE*.

Class	Top3 Similar Classes	Similarity	
		BERT$_{\text{p}}$	BERT$_{\text{p-}\mathbf{w}}$
Sea boat	Scow	0.94	0.91
	Row boat	0.93	0.91
	Canoe	0.93	0.91

Table 7: Similarity of Top 3 similar classes with *Sea boat* drops after applying the weighting approach.

G Ablation Study

Table 6 shows the results on 2-Hop with different thresholds τ introduced in Equation 5. We obtain the weighted average BERT$_{\text{p-}\mathbf{w}}$ by taking an input $\boldsymbol{h}$ from BERT$_{\text{p}}$ and learning MLP b_{ψ} with different τ (similar for BERT$_{\text{f-}\mathbf{w}}$). Then, we measure 2-Hop accuracy based on BERT$_{\text{p-}\mathbf{w}}$ (or BERT$_{\text{f-}\mathbf{w}}$). Note that BERT$_{\text{p}}$ and BERT$_{\text{f}}$ have different ranges of τ, since BERT$_{\text{f}}$ already has lower similarity between classes. This is because BERT$_{\text{f}}$ is trained with images (from seen classes) during fine-tuning, which makes BERT$_{\text{f}}$ more aligned with visual features and thus is more representative. We choose τ based on the ImageNet validation set of the seen classes.

Table 7 shows that the weighted average embedding BERT$_{\text{p-}\mathbf{w}}$ makes similar classes less similar. Originally, a class "Sea boat" has overly similar semantic representations with other type of boats (i.e. BERT$_{\text{p}}$). After applying our weighting approach, the classes become less similar (e.g. 0.94 to 0.91 between "Sea boat" and "Scow").

H Qualitative Results

H.1 Visual sections and clusters

We provide additional illustrations of visual sections and clusters of Section 3 in the main text.

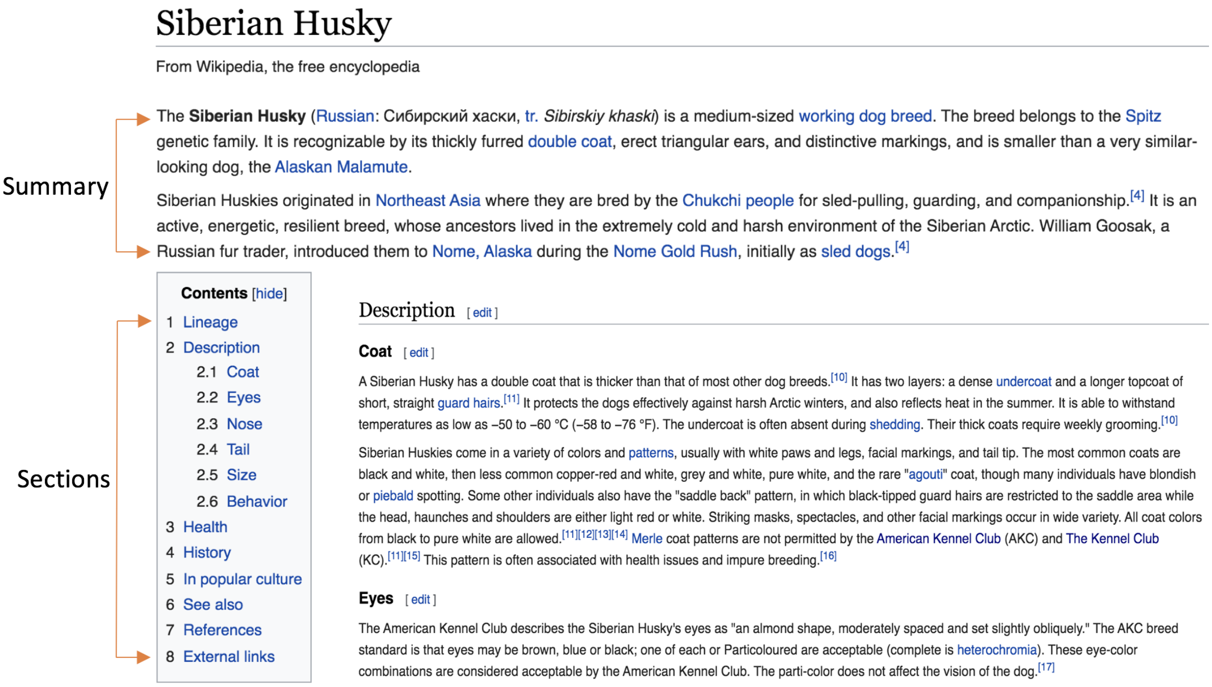
Figure 4 shows visual and non-visual sections in a Wikipedia page **Siberian Husky**. We note that the summary paragraph and sections such as *Description* contain visual sentences while sections such as *Health* or *History* do not. Similarly, Table 8 shows two clusters: the top cluster is visual, consisting of information about *hunting* and *preys* of animals while the bottom cluster includes *mythology* sentences not visually related.

H.2 On ImageNet

Figure 5 shows the qualitative results of our BERT$_{\text{f-}\mathbf{w}}$ and w2v-v2 on ImageNet. For each image, we provide its label and the Top 5 prediction by BERT$_{\text{f-}\mathbf{w}}$ and w2v-v2. While w2v-v2 is not able to differentiate the similar classes (e.g. Predicting "Scooter" as "Tandem bicycle"), our BERT$_{\text{f-}\mathbf{w}}$ can distinguish them. We also note that the Top 5 classes predicted by BERT$_{\text{f-}\mathbf{w}}$ are similar (e.g. "Grey whale" and "Killer whale"). This suggests that our approach maintains the order of similarity among classes but make their semantic representations more distinctive.

Clusters
· · · hunt shortly after sunset, eating small animals · · · · · · if food is scarce, it has been known to eat tomatoes · · · Tigers are capable of taking down larger prey like adult gaur · · · Tigers will also prey on such domestic livestock as cattle, horses, · · ·
Panda is a Roman goddess of peace and travellers · · · The Ibex is also a national emblem of the great ancient Axum empire. In Aztec mythology, the jaguar was considered to be the totem animal of · · · It is the national animal of Guyana, and is featured in its coat of arms · · ·

Table 8: K-means sentence clusters. The top cluster has *visual* information about *hunting* and *preys* while the bottom one contains *non-visual* description such as *mythology*.



Siberian Husky

From Wikipedia, the free encyclopedia

The **Siberian Husky** (Russian: Сибирский хаски, tr. *Sibirskiy khaski*) is a medium-sized working dog breed. The breed belongs to the Spitz genetic family. It is recognizable by its thickly furred double coat, erect triangular ears, and distinctive markings, and is smaller than a very similar-looking dog, the Alaskan Malamute.

Siberian Huskies originated in Northeast Asia where they are bred by the Chukchi people for sled-pulling, guarding, and companionship.[4] It is an active, energetic, resilient breed, whose ancestors lived in the extremely cold and harsh environment of the Siberian Arctic. William Goosak, a Russian fur trader, introduced them to Nome, Alaska during the Nome Gold Rush, initially as sled dogs.[4]

Contents [hide]

1 Lineage
2 Description
 2.1 Coat
 2.2 Eyes
 2.3 Nose
 2.4 Tail
 2.5 Size
 2.6 Behavior
3 Health
4 History
5 In popular culture
6 See also
7 References
8 External links

Description [edit]

Coat [edit]

A Siberian Husky has a double coat that is thicker than that of most other dog breeds.[10] It has two layers: a dense undercoat and a longer topcoat of short, straight guard hairs.[11] It protects the dogs effectively against harsh Arctic winters, and also reflects heat in the summer. It is able to withstand temperatures as low as −50 to −60 °C (−58 to −76 °F). The undercoat is often absent during shedding. Their thick coats require weekly grooming.[10]

Siberian Huskies come in a variety of colors and patterns, usually with white paws and legs, facial markings, and tail tip. The most common coats are black and white, then less common copper-red and white, grey and white, pure white, and the rare "agouti" coat, though many individuals have blondish or piebald spotting. Some other individuals also have the "saddle back" pattern, in which black-tipped guard hairs are restricted to the saddle area while the head, haunches and shoulders are either light red or white. Striking masks, spectacles, and other facial markings occur in wide variety. All coat colors from black to pure white are allowed.[11][12][13][14] Merle coat patterns are not permitted by the American Kennel Club (AKC) and The Kennel Club (KC).[11][15] This pattern is often associated with health issues and impure breeding.[16]

Eyes [edit]

The American Kennel Club describes the Siberian Husky's eyes as "an almond shape, moderately spaced and set slightly obliquely." The AKC breed standard is that eyes may be brown, blue or black; one of each or Particoloured are acceptable (complete is heterochromia). These eye-color combinations are considered acceptable by the American Kennel Club. The parti-color does not affect the vision of the dog.[17]

Figure 4: Visual sections on *Siberian Husky*.

Image / Label		Semantic Type	Top 5 prediction
	Tiger	BERT_{f-w}	Tiger, Tiger cat, Tabby, leopard, Jaguar
		w2v-v2	Tiger cat, Tiger, Cougar, Madagascar cat, Standard poodle
	Scooter	BERT_{f-w}	Scooter, Tandem bicycle, mountain bike, forklift, police van
		w2v-v2	Tandem bicycle, Scooter, forklift, mountain bike, police van
	Grey whale	BERT_{f-w}	Grey whale, Killer whale, Pelican, Sea lion, Sturgeon
		w2v-v2	Killer whale, Grey whale, Sea lion, Ice bear, Cocker spaniel
	Sports car	BERT_{f-w}	Sports car, Race car, Garden cart, Minivan, Limousine
		w2v-v2	Jeep, Sports car, ambulance, fire truck, taxi
	Printer	BERT_{f-w}	Printer, Hard disc, Polaroid camera, Slot machine, Chocolate sauce
		w2v-v2	Hard disc, Cannon, Printer, Thimble, Stethoscope

Figure 5: Qualitative results between **BERT**$_{\text{f-w}}$ and **w2v-v2** on ImageNet. For each image, we report Top 5 prediction. While **w2v-v2** is not able to distinguish similar classes (e.g. Predicting "Scooter" as "Tandem bicycle"), our **BERT**$_{\text{f-w}}$ differentiates them.