Semi-supervised Learning Of Jointly Aligned Concept Embeddings

Candidate number: LNGT0[1]

MSc Machine Learning

Supervised by: Professor Bradley Love and Dr. Brett Roads

Submission date: 13 September 2021

Abstract

We are interested in algorithms that learn like humans. Humans are natural multimodal learners, applying knowledge and skills gained from one task to many other tasks. Studies in multi-task machine learning conclude that this approach of learning from multiple input streams results in computer models that generalise better. We test this by simultaneously learning probabilistic embeddings of concepts for two domains - images and audio - from co-occurrence statistics of concepts in the Open Images and AudioSet datasets.

Alignment between domains is defined as knowing the mapping from one domain's embeddings to another; thus we further constrain the problem by inducing alignment during the learning process using the labels present in both datasets as semi-supervised input. The embeddings for both domains, as well as the alignment between them, are learned jointly. The extra constraint should reduce the hypothesis space, decreasing the chance of being stuck in local minima as is a known issue with learning of embeddings. The Maximum Mean Discrepancy statistic is also investigated as a means for increasing alignment.

As a measure of embedding quality, we compare the cosine similarity of concept pairs from our aligned embeddings with the similarity of pairs from three human-curated datasets containing human similarity judgement (WordNet, enhanced ILSVRC and MTURK-771). We find that semi-supervised alignment increases embedding quality of both domains when compared with WordNet, the largest of these human-curated datasets. In particular, the embedding quality of the smaller domain (AudioSet) is increased.

Acknowledgements

Special thanks are owed to the following:

- Professor Bradley Love, for the opportunity to work with the Love Lab.
- Dr. Brett Roads, for much help and guidance plus access to `love17` and lots of computing power.
- Professor Justine Cassell, who taught me how to write my first dissertation, without which writing this one would have been much harder.
- Dr. J. Jason Ward and Dr. Paul Voutier for their help with reading initial drafts.
- Dr. Chong Siew Meng (1950-2019), without whose influence I would not have done this degree. The opposition was thumped.

Code

Code is available at:

`https://anonymous.4open.science/r/spond-C00C/README.md`

Only the specific subdirectory `spond/experimental/glove` is my code; the rest of the repository is a pre-existing library developed by the Love Lab, which I extended.

Contents

Chapter 1

Introduction

To build a system that learns in a similar way to humans, we must first identify what we want to learn, and the features of human learning that we want to emulate. We then have to find an appropriate computational representation for the learned items, and an algorithm to perform the learning.

First, we consider how humans represent ideas internally. We take a definition from philosophy [ML21] and cognitive science [Pin07] of concepts as psychological entities that form an internal mental system. In both human and machine representation, the structure formed by the interconnection of these representations is significant.

In machine learning, we can describe concepts as statistical regularities in the world [RL20b]. These are often represented computationally as embeddings- real-valued vectors. Learning these statistical regularities should be done in a way that preserves the concept topology, so that concepts close together semantically should be close together in embedding space.

Different types of media (for example, text, images and audio) provide different presentations of the same concepts. If human experience can be considered to be the underlying generative process for these different media, we can expect that similar relationships should be found between concepts as expressed in these media. For example, pictures of apples are more likely to contain pears or oranges, than violins and buses, and documents containing the word "apple" are more likely to have the word "pear" in close proximity compared to the words "violin" or "bus"- so we would like embeddings learned from images and text to display the behaviour that "apple" is closer to "pear" than to "violin" or "bus". These co-occurrences of which concepts occur together constitute information that can be used to learn concept representations.

The human learning paradigm that we try to emulate is multi-task learning, where information from multiple streams is used for simultaneous learning. We will do this by learning embeddings to represent human-defined concepts from two sources of media simultaneously. We will use probabilistic embeddings, where each concept is represented as a distribution (i.e., two vectors, one for the mean and one for the variance), as the learned variances will give us information about the uncertainty of the embedding.

The process of learning embeddings is stochastic, highly dependent on initial conditions, and subject to many local minima due to the complex loss surface. Thus, we impose an alignment constraint during this learning process such that the two resulting embedding structures (one for each domain) where alignment is defined as knowing the correspondence from a concept in one system to its analogue in the other system. The additional constraint should have a regularising effect by reducing the size of the hypothesis space to favour aligned solutions, as well as inducing the systems of embeddings in each domain to have similar internal structures.

We hope that the resulting aligned multi-modally learned embeddings will be more similar to human representations than uni-modally learned embeddings, in keeping with [Rud17] which finds multi-task machine learning to generalise better. We test this by comparing pairwise similarity of our embeddings (for both the multi- or uni-modally learned cases) with human judgements of pairwise similarity. The ideal outcome is for jointly aligned embeddings to be superior to independently learned embeddings; that is, for the data from one domain to induce better quality in the embeddings learned for the other domain.

The input dataset comprises co-occurrence matrices constructed from the Google Open Images (19996 concepts) [Kuz+18] and AudioSet (526 concepts) [Gem+17] datasets. A full description of the dataset will follow in a later chapter; this dataset contains a total of 20292 concepts with an overlap of 230 concepts. The concepts present in both domains will be used to perform semi-supervised alignment. The two modalities are unbalanced in terms of number of concepts and it will be interesting to see if the domain with many concepts (Open Images) will help to induce better quality embeddings in the domain with fewer concepts (AudioSet).

Chapter 2

Background and existing work

In this chapter we present an overview of the main ideas pertaining to this project, whose aim is to learn aligned concept embeddings for multiple domains simultaneously, then evaluate the quality of those embeddings compared to unaligned embeddings. We outline the philosophical basis for concepts, and explore why the relationships between concepts are significant. We also briefly survey different types of embeddings and their characteristics. As our hypothesis is that multimodal learning will give better results than unimodal learning, we look at the reasoning behind this hypothesis. Lastly, we define alignment both in general terms and the specific terms of this problem, and consider ways that alignment may be achieved.

2.1 Concepts

There is a philosophical theory known as the "representational theory of the mind" [ML21] in which concepts are defined as psychological entities that make up an internal system of representation. The relationships between concepts may form part of their definition; that is, the concepts may be implicitly defined by their relationships to other concepts as well as having explicit properties of their own. The relationship between concepts is a key theme in this project, as it is these connections that define the structure of the concept network, which is key to the alignment problem.

The [ML21] definition is also a main definition used in cognitive science [Pin07]. In [PP96], concepts are related to human characterisation of categories of objects, noting that categories may overlap and have fuzzy boundaries. In [GR02], concepts are again defined not only by features (apple maps to "red or green or yellow in colour", "round in

shape") - the "external grounding" description of meaning, but also by their relationships to each other ("apple" is more like "pear" but less like "strawberry"). The "conceptual web" description states that a concept's meaning is defined by its place within the entire structure formed by all concepts.

In the field of computer science, there exist many frameworks of concepts defined in machine-interpretable ways. Some examples are Cyc [Len95] (an ontology of "common-sense" rules and concepts), WordNet (a lexical database of English words, with the inherent assumption that words and concepts are interchangeable) [Mil95], and the Google Knowledge Graph [Hei+20]. These store the connections between concepts like hierarchy and inference, not just names; in this field too, we see that these relationships encode valuable information.

There is precedent for this way of thinking in the psychological and cognitive science literature. [SC70] found that the connections between concepts (internal mental representations) should reflect the connections between the external (real world) representations; this was experimentally tested by querying subjects on the identification of US states by shape, and further confirmed by [GH73], testing the identification of faces. [GR02] found an algorithm that was able to use the internal relationships between concepts from disparate systems to find correspondence between those two systems.

[BE21] found that object representations in the human brain (which can be thought of as concepts - mental representations) are related to the co-occurrence statistics of objects occurring in images and described in language. They posit that objects occur in context, with certain objects occurring together more often than not, and that the brain possesses mechanisms to support this type of contextual knowledge. They found that brain response as measured by fMRI activity can be predicted by the co-occurrence statistics of objects within a visual scene. [SNG13] also found that the brain's response to scenes could be predicted based on clusters of co-occurring objects in that scene.

Therefore, research evidence suggests that concepts may be considered to embody observed phenomena or events in the world, and that the relationships between concepts are important. Any machine learning system aiming to learn concepts in a similar way to humans needs to take this into account.

2.2 Embeddings

In machine learning, embeddings are continuous real-valued vector representations of features. Language embeddings (built from textual input data) are the most commonly known type, but embeddings may be constructed from any data source. They are essentially a form of dimensionality reduction. For example, a one-hot representation of words in a corpus may have dimensionality in the millions, but converting these to word embeddings might reduce the problem dimensionality to several hundred. One of the earliest word embedding algorithms is `word2vec` [Mik+13], in which a neural network is used to learn embeddings from a corpus by projecting similar words to similar locations in the target vector space. This algorithm takes advantage of co-occurrence patterns (some words occur in the neighbourhood of other words). The resulting embeddings should reflect the properties of words, where words with similar meaning are close together in embedding space. The "Continuous Bag Of Words" (CBOW) variant uses a window of context (disregarding order) to predict the current word, and the Skip-gram variant uses the current word to predict the context window, weighted by distance from the current word. In both cases, we see that the context of the current word is important, pointing again to the relationship between the concepts being a key input for embedding creation.

The GloVe [PSM14] embedding algorithm is also based on co-occurrence statistics. The co-occurrence of a word pair is the number of times those words occur within a set context window of each other. The GloVe algorithm attempts to learn embeddings whose dot products give rise to the particular co-occurrence statistics of the corpus. These algorithms can be generalised to learn embeddings from any source of co-occurrence statistics, not just words; in [BE21], a set of object embeddings called `object2vec` was built from co-occurrence statistics from the ADE20K data set [Zho+16] of images labelled by human annotators, using the `word2vec` algorithm. In this project, we do the same for the GloVe algorithm, learning embeddings from co-occurrence statistics of non-textual inputs.

[MLS13] found that word embedding spaces have similar structure over different languages, even when those languages are linguistically quite far apart, like English and Vietnamese- another example of how the relationships between entities in the concept systems are significant.

The Laplacian eigenmap algorithm [BN03] is a geometrically motivated graph-based algorithm for learning embeddings. A weighted graph is created with a node for each concept/feature, and edges are added between nodes if they meet some definition of closeness (such as Euclidean distance, or being in the k-nearest neighbour set). The embeddings

are the eigenvectors of the graph Laplacian. This algorithm is also context-based but in a different way; it preserves information about the entire local neighbourhood geometry.

2.2.1 Probabilistic embeddings

Most algorithms for generating embeddings have significant stochasticity, as the learning process often involves minimising loss using stochastic gradient descent. The loss function landscapes are complex with many local minima, and the solution is sensitive to initial conditions. Therefore, runs with different random seeds will produce different embeddings; not just the individual embedding values will differ, but also the relationships between those embeddings. In a later part of this document we show examples of this.

All of the previously mentioned embedding generation algorithms result in point estimates, which are deterministic in the sense that each embedding is a single vector of numbers. We can generalise this idea to probabilistic embeddings, in which each embedding is represented by multiple vectors that are parameters of a distribution. For example, we may consider embeddings to be multivariate Gaussians with independent dimensions (diagonal covariance matrix), therefore representing each embedding with two n-dimensional vectors, for the mean and variance respectively. By learning the variance as well, we derive further information about each concept that can be used to quantify the degree of uncertainty of that concept in the system. When the embeddings are used, a sample is taken from the distribution of each concept whose embedding is desired.

[Chu+21] represent cross-modal (visual and language) embeddings as probability distributions in an embedding space common to both modes, and use the uncertainty represented by the learned variance in automated decision making. Including the uncertainty in their information retrieval task improved performance, as well as increasing the interpretability of the final embeddings. Looking ahead to the problem of finding correspondences between embeddings in multiple domains, we also see that variance can provide additional information for disambiguation during the mapping process.

[Zho+19] and [VM15] learn word embeddings by modeling each embedding as a probability density function. [Zho+19] learn bilingual embeddings in an unsupervised way by matching the densities of the two monolingual embedding spaces. This allowed them to achieve state-of-the-art results on linguistically distant language pairs without the special initialisation or complicated optimisation required by many unsupervised methods. [VM15] learn a mapping from individual words, as well as words in context, to a Gaussian distribution over a latent space. This mapping is such that the linguistic and seman-

tic properties of the words are preserved by the relationships between the distributions. Words that appear in similar context should have similar embeddings. In particular, they wanted entailment relationships[1] to be represented by the means and variances of the words' respective distributions. A diagram, reproduced below from [VM15], demonstrates the concept:

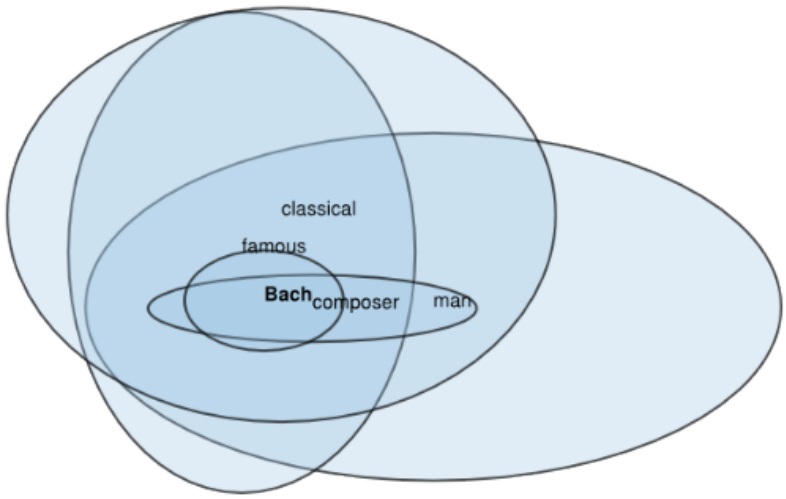

Figure 2.1: Figure taken from [VM15]. The ellipses represent the projection onto eigenvectors of the mixture means and variances of the respective words' distributions. The pattern is that items that entail other items should be substantially contained within the entailed items' ellipses. For example, Bach is a famous classical male composer, and thus the ellipse for Bach is contained (mostly) within the ellipses for all of the other words' classes.

[1] If there is an entailment relationship between A and B, A must imply B.

2.3 Machine learning paradigms

2.3.1 Supervised / unsupervised learning

Supervised learning requires external labelling of data to give the system information about the "correct" things to learn. For example, a category label for an image, or that the word "cat" in English corresponds to "kucing" in Malay. Unsupervised learning requires the learning system to extract regularities from the data without such guided input. There has to be some external input, for example, telling the system about the universe of available concepts, but there are fewer constraints imposed on the algorithm, which should generalise using only the data values. Semi-supervised learning falls in between, where we know "correct" labels or categories for only some items in our dataset.

Learning in humans comprises both types. Humans often learn by comparing their activities or ideas with known correct data, or by receiving feedback from an external agent about their correctness of their actions; this is supervised learning. Humans also learn in an unsupervised way, by making inferences from all the data available to them in different modalities such as sound, vision, and natural language.

2.3.2 Multi-task learning

Human learning often follows the multi-task paradigm, where we apply knowledge to one task that was obtained from learning related tasks. Most of us learned arithmetic in primary school, and we apply that to checking the transactions of our bank account, or to adding up the total at the supermarket till. Basic skills are built upon to allow more complex techniques to be learned.

The machine learning equivalent is known as multi-task learning, where more than one objective is minimised at a time, and training signals of one task can affect the signals of other tasks in the same ensemble. A synergistic effect is observed where models learned from multi-task learning tend to generalise better [Rud17]. By training on multiple tasks, the model can learn a more general representation of the pattern that underlies all those tasks. If this general representation is closer to the true universal generative representation, it will allow the model to generalise to novel tasks better (partially overcoming the bias-variance tradeoff).

By learning from multiple tasks simultaneously, we are using more data than we would use in learning only a single task. Training a model requires learning a good generalisation

for the task that ignores the noise in the data. As different tasks have different patterns of noise, learning multiple simultaneous tasks should allow learning of a more general representation. This reduces the risk of overfitting to only one task. There is also a regularising effect as multi-task learning adds an inductive bias that makes a model favour some hypotheses over others. In the context of our embedding alignment problem, the learning of embeddings in one domain acts as an auxiliary task for the learning of embeddings in the other domain, with each domain acting as an inductive bias for the other. The end result should induce the model to favour hypotheses that explain both tasks, with the hope that this model is more representative of the "ground truth" of the real world that generates their co-occurrence statistics.

2.4 Alignment and ways to achieve it

2.4.1 Some definitions of alignment

Alignment as mapping two spaces to a single latent space

In its simplest form, alignment requires learning to map between two vector spaces. The notation from [WPM11] is used here to state the problem, as we find it particularly clear.

If there are two datasets X and Y whose instances all lie on the same underlying manifold Z, one version of the alignment problem requires finding functions f and g such that $f(x_i)$ is close to $g(y_j)$, for some problem-specific definition of distance.

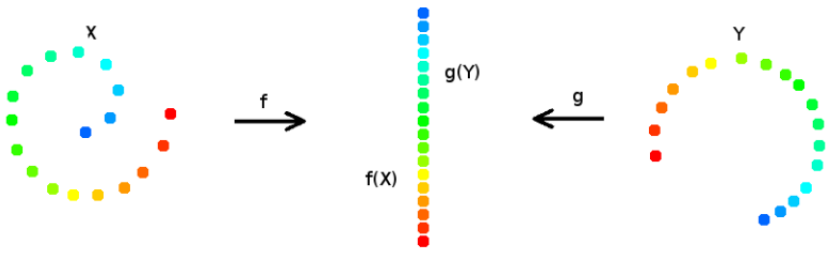

Figure 2.2: Figure taken from [WPM11]. The $\boldsymbol{X}$ and $\boldsymbol{Y}$ spirals denote the two datasets, with the center line showing the data points embedded into the shared space, with local similarity relations being preserved. f and g denote the functions mapping $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively into the shared space.

If $f(x_i) = g(y_j)$, then x_i and y_j are in correspondence. If we know ahead of time that x_i and y_j are analogous points in their datasets, then we can provide this information to the algorithm that is trying to infer f and g. The union of the ranges of f and g is then the joint latent space. This can be generalised to more than two datasets.

This however is a definition that maps both domains into the same space.

Alignment as learning mappings from one domain to another

In the previous definition, alignment requires that both domains be mapped to a single latent domain. This can be useful for certain problems, for example, to find a single language-independent representation of the data in multilingual related corpora.

There is another definition for alignment, which requires only that domains can be mapped to each other directly.

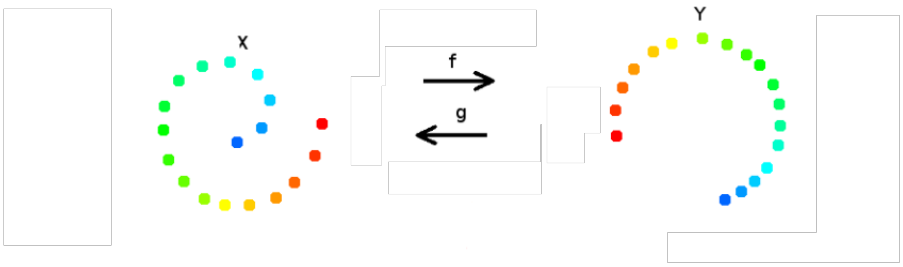

Figure 2.3: The $\boldsymbol{X}$ and $\boldsymbol{Y}$ spirals still denote the two datasets, but this time f and g denote the functions mapping $\boldsymbol{X}$ and $\boldsymbol{Y}$ respectively to each other, instead of into some shared space.

We wish to find mappings $f(\boldsymbol{X}) \to \boldsymbol{Y}$ and $g(\boldsymbol{Y}) \to \boldsymbol{X}$ where the spaces $\boldsymbol{X}$ and $\boldsymbol{Y}$ have similar structures. This is slightly different from the definition of alignment in the previous section. This definition may be used when it is not clear what the dimensionality of the latent space should be.

We will use this definition of alignment in our problem, as we indeed do not know what an appropriate dimensionality should be for a latent space that represents data from both domains.

Second order isomorphism

Returning to an idea first raised in [SC70], we want second order isomorphism- we want to know the functional relationships between clusters of concepts over our two modalities. Even if there is no structural resemblance between a concept and that concept's representation, the same structure should be observed between the relationships between the concepts and the representations. As stated by [GR02], the meaning of a concept is tied to its relationships to other concepts in the same network. Similarity relationships between concepts within the same system can therefore be used to map between systems; using only these relationships, [GR02] found an algorithm, ABSURDIST, that was able to translate between two such networks. Without constraints on the systems, or even on the number of concepts in each system, ABSURDIST was able to find translations using only similarity matrices created by some external agent (for example, a German-English bilingual human). ABSURDIST found that, while within-system relationships are enough to find a translation, this translation can be made more robust to noise by adding external, extrinsic information about correspondences (for example, higher weights for certain correspondences). This is a concrete example of why the relationships between concepts are important; because they can be used to perform alignment.

[GR02] found that ABSURDIST's mappings were better for systems that had a greater number of elements. This is because such a system has more similarity relations, and each similarity relation provides a constraint to uniquely identify each member. Therefore, systems with more elements are more constrained.

Applications of alignment problems

Some other examples of alignment problems spanning a range of applications are given below.

- Biological manifold alignment: [AK18] applies Generative Adversarial Networks [Goo+14] to the problem of alignment of cell correspondence between cytometry batches (knowing which cells generated which biological results).
- Neural style transfer: [Zhu+17] takes as input differently styled source and target image sets and applies adversarial network techniques to translate images from the source set style into the target set style.

- Bilingual lexical induction: [Con+18] applies unsupervised alignment between monolingual word representations to derive a dictionary between those languages.
- Deep multimodal embedding: [SLS15] relates information from three modalities-point-cloud, natural language and manipulation trajectory data, to teach a robot arm how to manipulate new objects.

2.4.2 Methods for concept alignment

In this section, the following notation applies:

$\boldsymbol{X}$ denotes embeddings in the source domain,

$\boldsymbol{Y}$ denotes embeddings in the target domain,

$\boldsymbol{W}$ denotes a transformation matrix.

Regression and related models

Regression models are the simplest form of alignment model. They only require learning a transformation matrix $\boldsymbol{W}$ from one domain to another (minimising the mean squared loss $||(\boldsymbol{W}\boldsymbol{X} + \boldsymbol{b}) - \boldsymbol{Y}||_2^2$) which can then be applied to a new source vector to map into the target space. [MLS13] uses this to find a "translation matrix" that can map from word embeddings in one language space to another. Orthogonal models constrain the learned $\boldsymbol{W}$ matrix to be orthogonal, which is appropriate if the angles between embeddings are more important as a measure of similarity than the distances between them. Various standard preprocessing steps may be applied like normalisation to unit norm / mean centering, decorrelation to have unit variance, or SVD for dimension reduction.

Maximum margin models are similar to the support vector machine [CV95] in that they balance increasing the weights from matching pairs with reducing the weights learned from known generated noise pairs. An example loss function (reproduced from [KA20]) is:

$$\sum_{i=1}^{n}\sum_{j\neq i}^{k}\max\{0,\gamma-\cos(\boldsymbol{W}\boldsymbol{e}_i^s,\boldsymbol{e}_i^t)+\cos(\boldsymbol{W}\boldsymbol{e}_i^s,\boldsymbol{e}_j^t)\}$$

with $\boldsymbol{e}_i, \boldsymbol{e}_j$ being the ith and jth of n embeddings,
$\boldsymbol{W}$ being the transformation matrix,
k being the number of noise pairs,
and γ being the margin parameter.

This is different from regression models which rely largely on minimising the mean-squared error. [LDB15] found that this form of loss reduced the effect of hubs that can plague regression and orthogonal algorithms. Hubs are embeddings with high similarity to all other embedding vectors in the space, due to entities that are too common (such as words that are very frequent in a corpus) or simply as an artifact of the transformation where many features get mapped to a small region of embedding space spuriously. [Jou+18] introduce another metric to reduce hubness, cross-domain similarity local scaling, that incorporates the mean of k-nearest neighbour distances of points in the target space to capture the neighbourhood geometry.

Point set registration, most commonly found in computer vision, is the alignment problem applied to sets of points in 2 or 3 dimensions. [Zhu+19] reviews many current techniques; one key difference between this problem and the general alignment problem is that the points are known to lie on a 2- or 3-dimensional manifold, and this is quite a significant constraint. We cannot make any such assumptions about the dimensionality of our embeddings.

Manifold alignment models

As stated in [WPM11], manifold alignment is a type of constrained simultaneous dimensionality reduction with the aim of finding a low-dimensional embedding for all input datasets that preserves the topology of correspondences between them. The datasets may have disjoint features. The data may be of very high dimension, but if all the data points lie on a low-dimensional manifold, this manifold may be learnable.

This requires the multiple datasets to be representable by a shared underlying structure. It may be convenient to learn the mappings between datasets without ever formalising this shared structure, or it may be useful to find common features. For example, in language

translation it may be enough to know the mappings from words or phrases in one language to the equivalent entities in another language. However if the problem is multilingual information retrieval, it may be more useful to express the translations of different documents as a single underlying joint representation. Our specific problem is learning aligned embeddings from co-occurrence data that represent the statistics of how concepts occur in different human-created media. If we consider the ultimate underlying generative process of all these media to be "the real world", it is plausible that the embeddings could have some underlying shared joint representation, but we do not know anything about this representation, particularly the dimensionality of the underlying process.

Some existing dimensionality reduction techniques that can be used in alignment are the Isomap [TSL00], which reduces dimension while preserving distances between points; locally linear embedding [RS00], which reduces dimension while keeping distances the same between local neighbourhoods; and the previously mentioned Laplacian eigenmap [BN03], which approximates the manifold by the adjacency graph derived from the embeddings. These algorithms all try to find a low dimensional representation of a single dataset, and manifold alignment simply uses these or similar algorithms to find embeddings for multiple datasets simultaneously. Without correspondence information, manifold alignment will result in independent embeddings for each input dataset, but if direct correspondence information is provided, or a means for inferring it, manifold alignment will use this to constrain the embeddings to be aligned. Manifold alignment considers each individual dataset to be part of one larger dataset whose range includes the mapped values of all other datasets.

In [GJB18], an approach to aligning two sets of existing embeddings (learned separately from the alignment procedure) using a combination of Procrustes analysis and the Wasserstein distance is described. Procrustes analysis is normally used to learn an affine transformation between two sets of points with a known correspondence. If we consider $\boldsymbol{X} \in \mathbb{R}^{n \times d}$ (n vectors of dimension d) and $\boldsymbol{Y} \in \mathbb{R}^{n \times d}$ (another set of vectors of the same size), the Procrustes linear transformation is the solution to the following:

$$\underset{\boldsymbol{W} \in \mathbb{R}^{d \times d}}{\operatorname{argmin}} \quad ||\boldsymbol{X}\boldsymbol{W} - \boldsymbol{Y}||_2^2$$

[Goo91] used this to analyse two-dimensional shapes, where the shapes are considered to be the same if by application of rotation, translation and isotropic scaling, one can be transformed to the other. [MLS13] applied this technique to learn linear mappings between word embeddings in different languages using a bilingual dictionary.

The Wasserstein distance, also known as the Earth Mover Distance, is the solution of the following optimisation problem

$$\begin{aligned} &\underset{\boldsymbol{P}\in\mathscr{P}_n}{\operatorname{argmin}} \quad ||\boldsymbol{X}-\boldsymbol{P}\boldsymbol{Y}||_2^2 \\ \text{where} \quad &\mathscr{P}_n = \{\boldsymbol{P}\in\{0,1\}^{n\times n}, \boldsymbol{P}\mathbb{1}_n = \mathbb{1}_n, \boldsymbol{P}^T\mathbb{1}_n = \mathbb{1}_n\} \end{aligned} \tag{2.1}$$

where $\boldsymbol{P}$ denotes a permutation matrix. It enforces a one-to-one mapping from the rows of $\boldsymbol{X}$ to $\boldsymbol{Y}$.

[Zha+17] used this distance measure as a minimisation objective to perform automated translation without supervision. By using the Wasserstein distance as a measure of closeness between the source and target embedding spaces, they formulate a system in which minimisation of this distance draws the two distributions closer.

Graph matching and graph similarity methods

A system of interconnected concepts can be represented as a graph whose nodes represent concepts and whose edges represent relationships between concepts. For example, in [YM05], such a system is represented as a directed graph with N nodes whose edges are connected with labels from the set S representing all combinations of possible relationships between any two nodes. Examples of such relations are "Similar to", "Is-A" or other hyponymic / synonymic relationships, or weights corresponding to the degree of co-occurrence of the two concepts. [YM05], a sequel to [GR02], further describes how alignment of two conceptual systems may be formalised as matching two graphs.

The general graph isomorphism problem, that of telling if two graphs represent the same structure, is NP-hard [GJ90]. As cited in [YM05], much work has been done in the field of finding approximate isomorphisms; finding a function $s(.)$ such that for two graphs G_1 and G_2, the distance between $s(G_1)$ and $s(G_2)$ is minimised, where s is a problem-specific distance measure. There are polynomial-time algorithms for certain subtypes of graphs or trees, but in general heuristic algorithms are the only practical possibilities.

The ABSURDIST II algorithm described in [YM05] creates a matrix of feasible translations between the two concept graphs which is iteratively updated based on the similarity between distances between elements of the system. At the end of the iterations, the output is a correspondence matrix that describes the strength of correspondence between elements of the source and target sets; this is used to create the mapping from source to target items.

The `torch-two-sample` Python library and its corresponding paper [DK17] introduce smoothed versions of some graph-based similarity measures, such as the Friedman-Rafsky test [FR79] and a version of k-nearest neighbours [DK17], that can be used as loss functions for learning similar graph structures. These are not actual graph isomorphism methods as they do not directly provide an alignment mapping between the two graphs; rather, they aim to provide distributional similarity metrics based on graph-based properties. These are converted to smooth functions by taking the statistics to be expectations of a probability distribution, thus allowing implementations to be used as loss functions trainable by backpropagation. Further details may be found in the appendix (A.1).

Generative adversarial networks

The "generative" part of a generative adversarial model is a function (usually learned by a deep learning network) that outputs samples from a distribution indistinguishable from a particular source dataset. A discriminator model is then trained against the generated outputs so that it learns to tell the difference between real and synthetic outputs; this is the "adversarial" part. By alternating the training of these two models, the generator learns to produce better and better samples. In effect, the generator is learning to produce samples from the manifold on which the input data lies. This model was first introduced in [Goo+14] and while the main objective is usually to generate new samples from a given distribution, there are examples where it has been used for alignment.

The Manifold Alignment GAN [AK18] attempts to alleviate some of the problems with traditional GANs, one of them being that generated items can fool the discriminator at a batch level because the two manifolds are superimposed, rather than being aligned. When the manifolds are superimposed rather than aligned, it is as if the "edges" of the manifolds match, but the points within may be in any position (see Figure 2.4.2). If we were to overlay the manifolds, the corresponding points would not overlap. There are an exponential number of possible mappings that result in overlapping but unaligned manifolds.

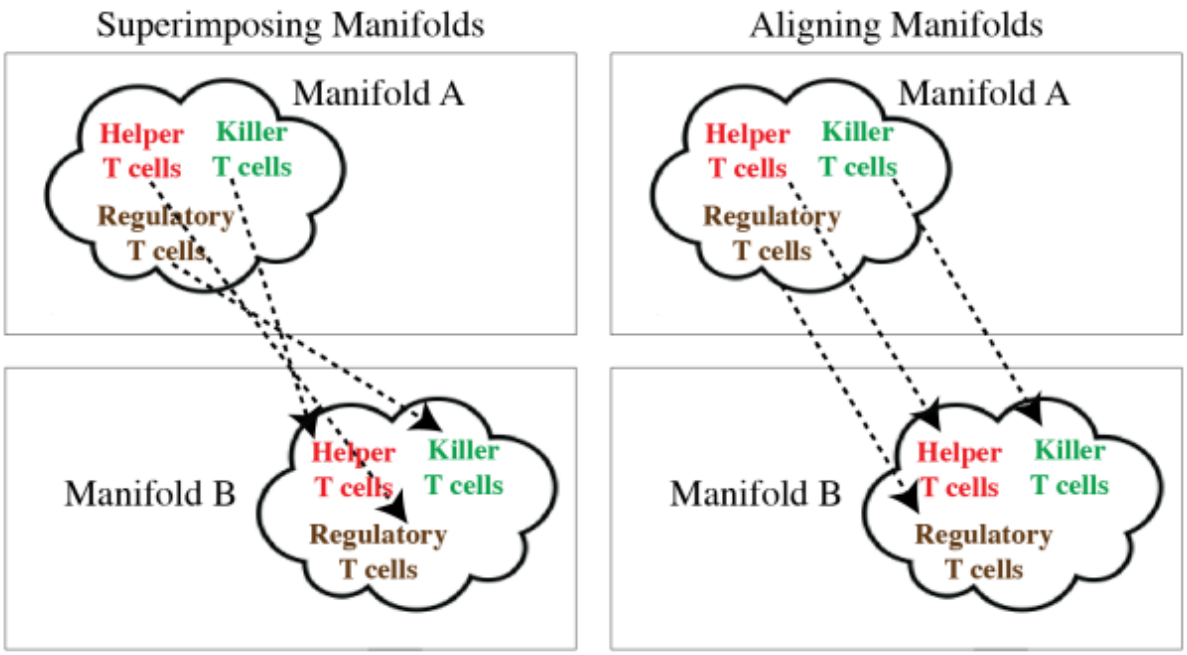

Figure 2.4: This diagram taken from [AK18] illustrates the conceptual difference between datasets that are merely superimposed (left panels), and datasets that are aligned (right panels). The MAGAN attempts to find pointwise correspondence between cells in one cytometry batch and another.

The MAGAN introduces a correspondence loss that measures the distance between a data point and its mapped image in the other domain, as well as the reconstruction loss (difference between an original point and its reconstructed image after going through both generators).

CycleGAN [Zhu+17] is not specifically an alignment model, but contains useful ideas for our purposes. CycleGAN is intended to provide image-to-image translation, where given two unordered image collections, one can be "translated" into the style of the other. For example, transforming pictures in the style of Monet into photographic style images. CycleGAN contains two GANs; one learns a mapping f from the input set $\boldsymbol{X}$ to the target set $\boldsymbol{Y}$, and one learns the mapping g which maps $\boldsymbol{Y}$ to $\boldsymbol{X}$. A key part of the CycleGAN model is the cycle consistency loss, which for the domain $\boldsymbol{X}$ is $f(g(\boldsymbol{Y})) - \boldsymbol{X}$ (there is an equivalent cycle consistency loss going the other way). This loss is intended to induce f and g to be consistent with each other; the MAGAN model also uses this loss, calling it "reconstruction loss". The cycle consistency / reconstruction loss is intended to reduce

the possibility of the learned mapping distribution matching the output distribution, but individual inputs not being mapped to individual outputs. This is a different way of expressing the alignment problem.

Certain types of datasets are better suited as inputs to GANs. Datasets with large numbers of object classes are not suited to GANs, which tend to underestimate the entropy in the distribution [Sal+16]. Many GAN implementations take images as input, and generally the input datasets are very large, providing many samples from which to learn the appropriate distribution. Our particular dataset could be considered to have only one stimulus per concept, if we take all concepts as end nodes in the taxonomy tree. Additionally, the salient features of images are more likely to lie on 2- or 3-dimensional manifolds. We do not know the appropriate dimensionality for the distribution that would represent any underlying manifold the concept embeddings may lie on; if the dimension of the GAN's latent space is not adequate, it will not be able to explore the sample space well, and learning will not occur.

2.5 Summary

Regression or other similar solely linear algebra-based techniques are not suitable, as our source and target embeddings are of very different size (19996 concepts vs 526 concepts with an intersection of 230 concepts). While there is a one-to-one mapping between the concepts in the intersection of source and target domains, the size of this intersection set is small relative to the number of concepts in each domain, and many concepts would be unaccounted for.

Manifold alignment techniques dependent on mapping both domains to a shared latent space are also not considered, as we do not know anything about the dimension of this space and the wrong choice of dimensionality would greatly affect any algorithm chosen.

Graph matching, being NP-hard, is not likely to be feasible as one of our domains has 19996 concepts forming a complex network. Any techniques from computer vision / point set registration are unlikely to work because of the previously mentioned constraint that most of those points will lie on a 2- or 3-dimensional manifold.

Generative adversarial networks work best when the input datasets meet certain criteria (outlined in the previous section), which ours do not. However, some of the loss functions used in various GAN architectures are of interest to us, and we will make use of them when designing our approach.

Chapter 3

Methodology and implementation

3.1 Datasets

Our input data comprises datasets from two modalities.

Open Images: The Open Images dataset [Kuz+18] collated by Google consists of approximately 7.3 million (7337077) annotated images tagged with concepts that occur within them. Images and concepts are given IDs. We parse the file that logs what concepts are present in which images to build a co-occurrence matrix of the number of times concepts occur in images over the whole dataset. This is exactly analogous to the co-occurrence matrix used in [PSM14] from which GloVe embeddings are derived. The dataset also contains other annotations like bounding boxes and hierarchy, but these are not used.

AudioSet: The AudioSet dataset [Gem+17] consists of 22160 annotated sound clips, where clips are given IDs and labelled according to which concept IDs are present in them. As this dataset is also collated by Google, the label IDs are identical and the concept names are almost the same as in Open Images (a small amount of preprocessing had to be done to match them exactly). As with Open Images, a co-occurrence matrix is created from the annotation data.

In total there are 20522 labelled concepts; 19996 from Open Images and 526 from AudioSet, with an intersection of 230 concepts present in both datasets. The concept labellings are both human- and machine-generated. They are not always accurate. For example, the image in Figure 3.1 below has been labelled, by a human annotator, with the terms "Tortoise" and "Sea turtle". As these two terms represent distinct species (one land dwelling and the other sea dwelling), the tagged object cannot be both. In the case of this image it is presumably because the human annotator could not determine the type

of animal represented by the statue next to the person. This pattern persists throughout the image library, where images are tagged with labels that are clearly related, but not always accurate. The images are consistently mis-labelled; the "Tortoise" / "Sea turtle" mismatch appears many times. For the purposes of this study this is less significant, but if we were trying to relate the relationships found by the experiments back to real-life data, it might be important. Nonetheless, the co-occurrence statistics capture the relationships that humans think exist, even if these are inaccurate. It is a philosophical point as to whether these are accurate in a different way- they capture an association that humans have for these concepts, even if that association is factually incorrect.

Figure 3.1: An image with at least one mislabelled concept: The fountain statue is labelled as both "Tortoise" and "Sea turtle"; it cannot be both.

In our experiments, there are no links between any concepts and any entities in the real world. The embeddings are learned completely independently of their associated labels and are numbered by the index they occur in the co-occurrence data. For example, the first index in the Open Images co-occurrence matrix is for "Sprenger's tulip", which has label `/m/0100nhbf`, but neither of these strings (the human name or the machine ID) are used anywhere in the learning system. Any meaning ascribed to relationships between embeddings must be inferred from re-mapping the indexes back to the concept names. If the algorithm generates embeddings for concepts with indexes 15268 and 5296 that are nearest neighbours in embedding space, we will not know until we perform this re-mapping that the concepts 15268 and 5296 are "Coffee" and "Tea" (in which case they are related),

or that they are "Sprenger's tulip" and "Ferret" (in which case they are not related).

The co-occurrence matrices for the Open Images and AudioSet datasets, as well as the concept name to label mapping files, make up the dataset for these experiments. This dataset should represent the statistical distribution of co-occurrence of concepts in the two modalities. The co-occurrence for a pair was incremented by 1 if both items in the pair were present in an image or audio clip. The raw data files also contained a confidence score for the degree of certainty of the labelling, but this was not used.

3.2 Embedding choice: Probabilistic GloVe

The dimension of the co-occurrence matrices is high, particularly the Open Images matrix which is 19996 by 19996 (AudioSet is a more manageable 526 by 526). However, they are extremely sparse, and thus ripe for dimensionality reduction. Since the GloVe learning algorithm uses only the non-zero items in the co-occurrence matrix, it is a suitably efficient choice. Also, its computational complexity depends on the number of such non-zero items rather than on the size of the co-occurrence matrix. It produces reasonable clustering, where concepts that are close in embedding space are also close in semantic space. GloVe embeddings are also stable, by the metric used in [BKM20]- the amount of overlap between nearest neighbours of an embedding for different runs. As the GloVe embedding is trained by stochastic gradient descent, the embeddings resulting from different random seeds will be different.

We modify this model to be probabilistic. The probabilistic/deterministic distinction refers not to the learning algorithm (as any algorithm trained by stochastic gradient descent will have a probabilistic component), but to the embedding representation. The original GloVe embeddings are represented by point estimates for each concept (thus a single vector for each concept); in our implementation, each probabilistic embedding represents a multivariate Gaussian distribution with independent dimensions (diagonal covariance matrix). We hope that the resulting learned variances will give us some knowledge of the degree of confidence we can have in each embedding, and also that learning two parameters for each embedding will help with alignment by providing more information for disambiguation.

A probabilistic embedding is a sample from the following distribution:

$$\mu + \sigma\mathcal{N}(0, 1)$$

Both μ and σ are learned parameters. In practice, σ is further parametrised as follows,

to enforce positivity:

$$\sigma = \ln(1 + \exp(\rho))$$

Therefore the full equation is:

$$\mu + \ln(1 + \exp(\rho)) \cdot \mathcal{N}(0, 1) \tag{3.1}$$

3.2.1 Implementation

The probabilistic version of the GloVe algorithm was implemented in Python using the PyTorch [Pas+19], PyTorch Lightning [Fal19] and Pyro [Bin+19] libraries. A custom neural network layer was implemented which took a co-occurrence matrix as input, and learned probabilistic GloVe embeddings by backpropagation through the GloVe loss function. Each mini-batch represented a random sampling of the rows and columns of the co-occurrence matrix, with all rows and columns being used over the course of one training epoch.

The GloVe loss function from [PSM14] is reproduced here:

$$\begin{aligned} L_{glove} &= \sum_i \sum_j f(X_{ij})(\boldsymbol{w}_i^T \boldsymbol{w}_j + b_i + b_j - \log X_{ij})^2 \\ f(x) &= \begin{cases} (x/x_{max})^\alpha & \text{if} \quad x \leq x_{max} \\ 1 & \text{otherwise} \end{cases} \end{aligned} \tag{3.2}$$

- The values of hyperparameters $\alpha = 0.75$ and $x_{max} = 100$ are taken from the original paper, [PSM14].
- The $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$ are samples of the current probabilistic embeddings represented by equation 3.1.
- The X_{ij} are co-occurrence statistics of the ith and jth concept.

The Pyro library provides backpropagation through the random sampling, as described in [GBC16] (section 20.9). The next section describes how these probabilistic embeddings were verified.

3.2.2 Validation

In this and following sections, the notation 1.234 ± 0.56 for an item denotes that the mean of N (usually 10) samples of the item observations is 1.234 and the standard deviation is 0.56.

The probabilistic embeddings were validated as follows:

- Train embeddings learned from Open Images and AudioSet co-occurrence data individually without regard for alignment between domains.
- For each run of each domain, set a different random seed. 10 seeds were used to get 10 instances of embeddings per domain.
- The following learning parameters were used:
  - Embedding dimension of 6. This choice was heuristic: the test implementation of GloVe embeddings implemented in PyTorch had 1 million unique tokens and dimensionality of 300, to keep the same ratio of effective tokens to dimensionality, 6 was the closest integer.
  - Mini-batch size of 500.
  - The Adam [KB17] optimiser, with learning rate 0.01.
  - 250 epochs of training for Open Images and 2000 epochs for AudioSet. This was enough to ensure convergence of the GloVe loss decreasing to asymptotic levels.
- No hyperparameter tuning was done; as this is not a supervised learning problem, we measure convergence only by the decrease in GloVe loss (equation 3.2) as there is no validation set to cross-check with.

This resulted in 10 sets of probabilistic embeddings for each domain. These embeddings are stochastic in two senses; one source is stochasticity in the learning algorithm leading to 2 runs with different seeds converging to different mean embeddings, and the other is caused by each embedding being a sample from a multivariate Gaussian distribution.

It is interesting that AudioSet needed many more epochs to converge. This is consistent with an earlier point mentioned in [GR02] that systems with more concepts are easier to learn as they are more constrained.

Clustering

As a sanity check, the t-SNE (t-Distributed Stochastic Neighbor Embedding, [MH08]) algorithm was used to reduce the dimensionality of the embeddings from 6 to 2, and the resulting points plotted for the top 300 most frequent concepts in each domain (measured by number of occurrences in images or audio clips). As we expected, there are clearly visible concept clusters. While t-SNE adds a further level of stochasticity during the dimensionality reduction process, two runs of t-SNE on the same input data with the same t-SNE random seed set, will produce the same result. Therefore, we can assume that the stochasticity introduced by t-SNE is accounted for.

The intent behind this qualitative analysis is only to confirm that the implementation is correct and converges to embeddings that have meaningful semantic relationships. This technique of using t-SNE to verify embedding quality is used in similar experiments, for example, [BE21].

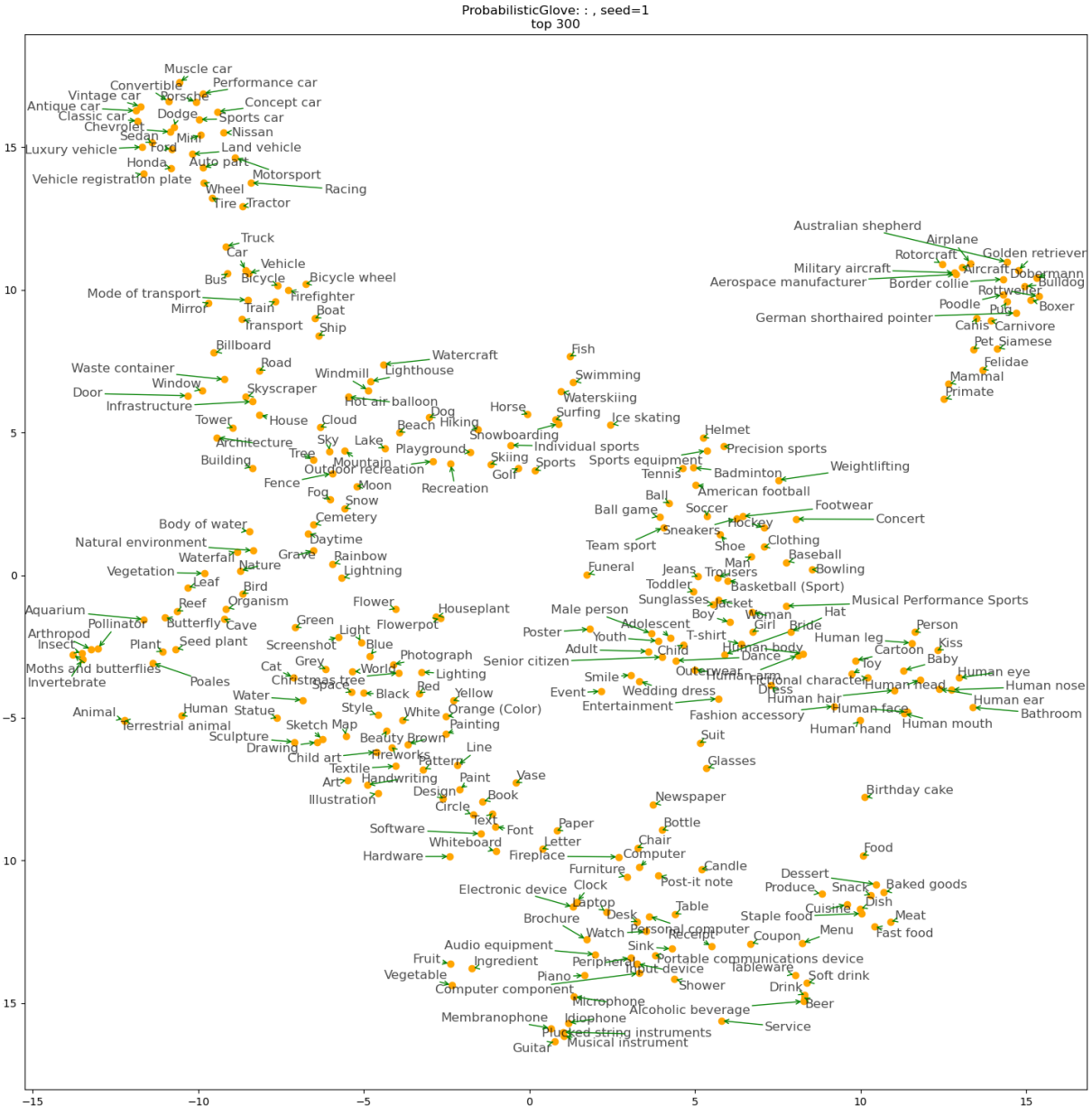

Figure 3.2: Open Images top 300 most common concepts. There are clearly visible clusters corresponding to, amongst other things, different types of dog, different types of vehicle, and different types of human body part.

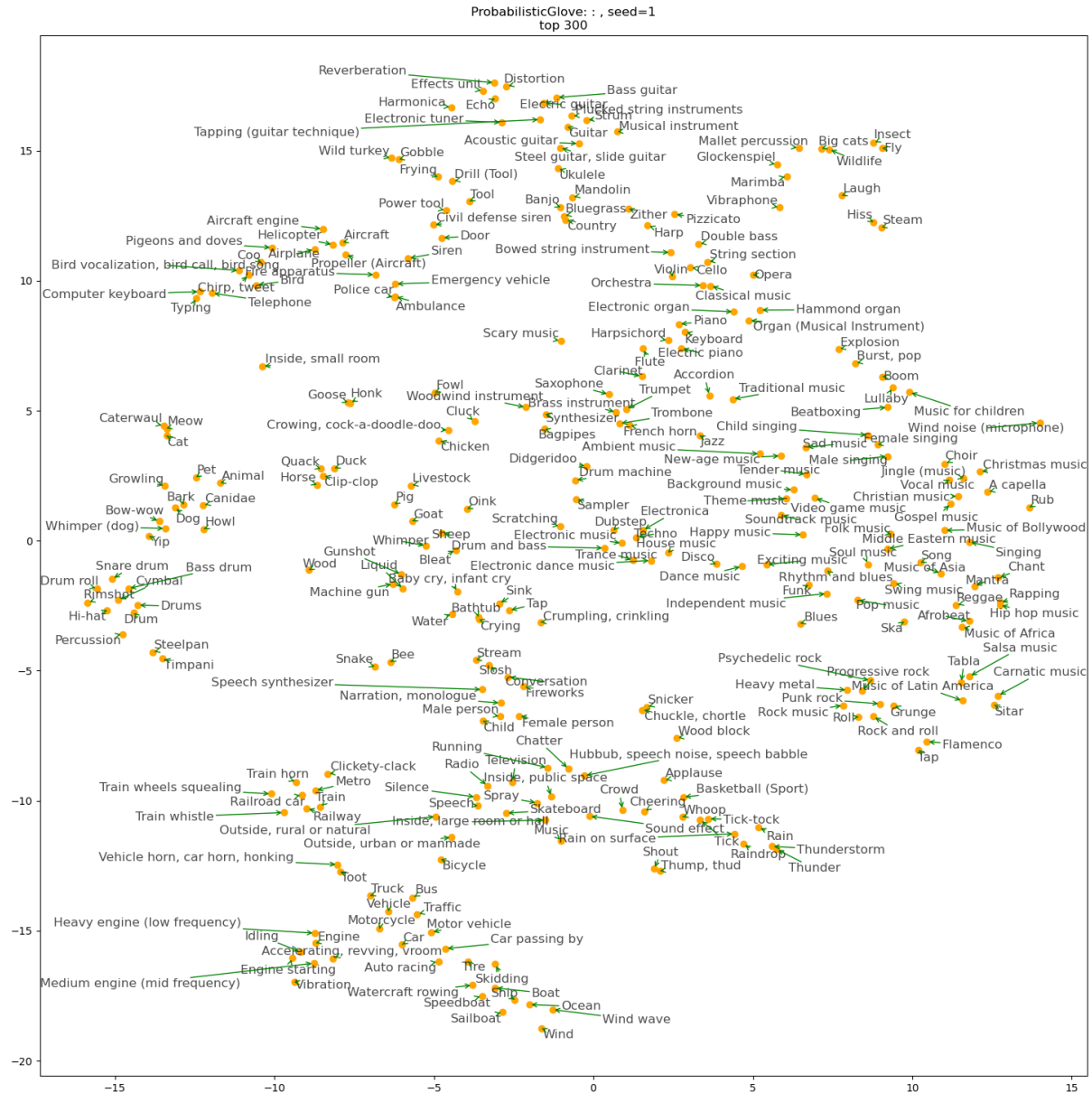

Figure 3.3: AudioSet top 300 most common concepts. Again there are clearly visible clusters comprising, for example, sounds made by water, sounds made by percussion instruments, and sounds made by dogs.

The following plots of the Open Images top 300 concepts (by frequency) t-SNE plots annotated to show some of the clusters, for random seeds 1 and 2, show that roughly the same concepts cluster for each run. They also reveal a characteristic of our alignment problem: **The cluster arrangements relative to each other across runs are**

not consistent. Ideally, we would like the different clusters to have the same spatial arrangement across runs. As described in [SC70], we would like to identify second order isomorphisms in the data; not just the clusters, but the relationships between the clusters. The variability from run to run suggests that the available data does not uniquely identify a single system of second-order isomorphisms. We attempt to inject additional constraints in a later experiment.

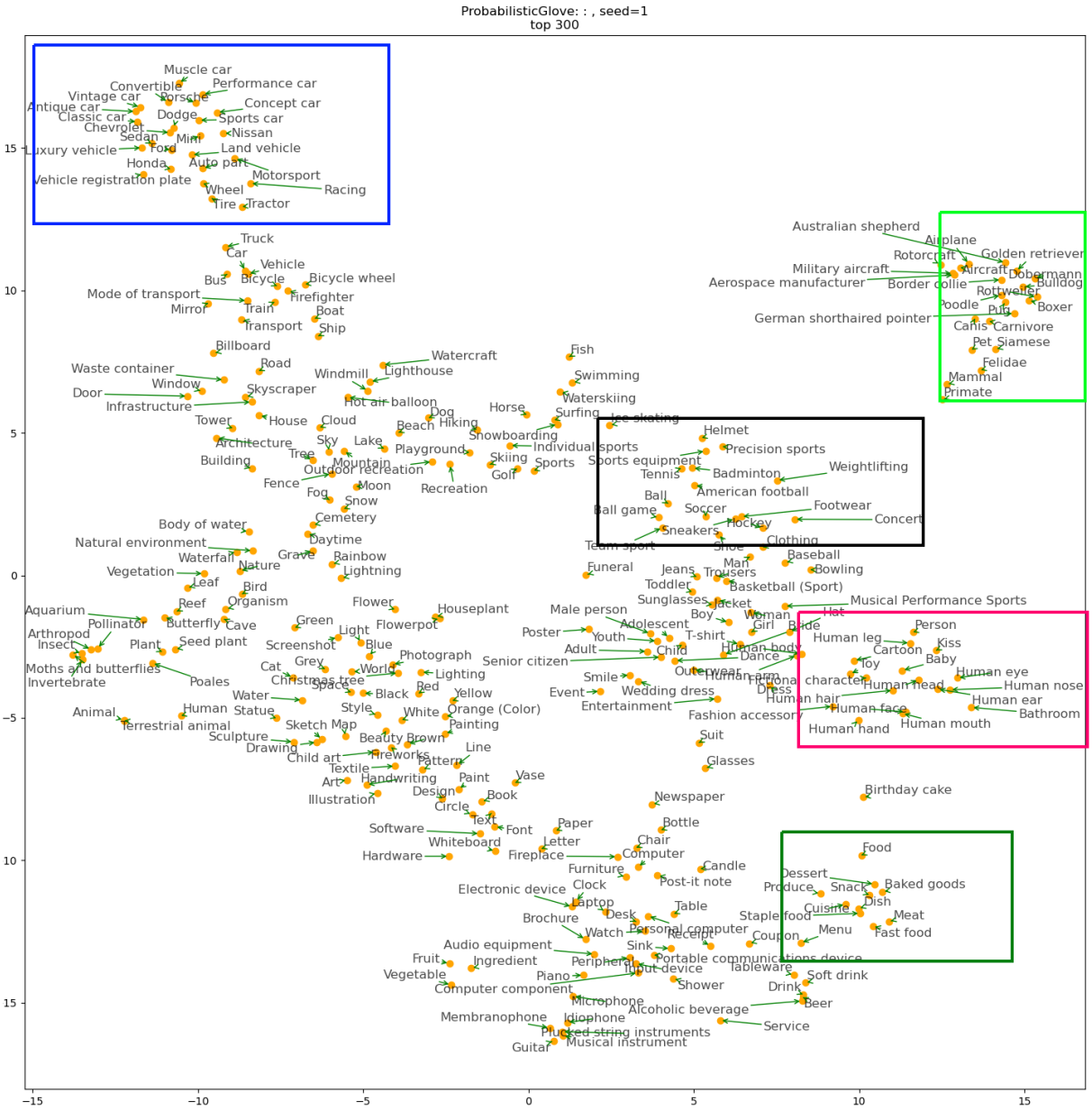

Figure 3.4: Coloured boxes indicate clusters.

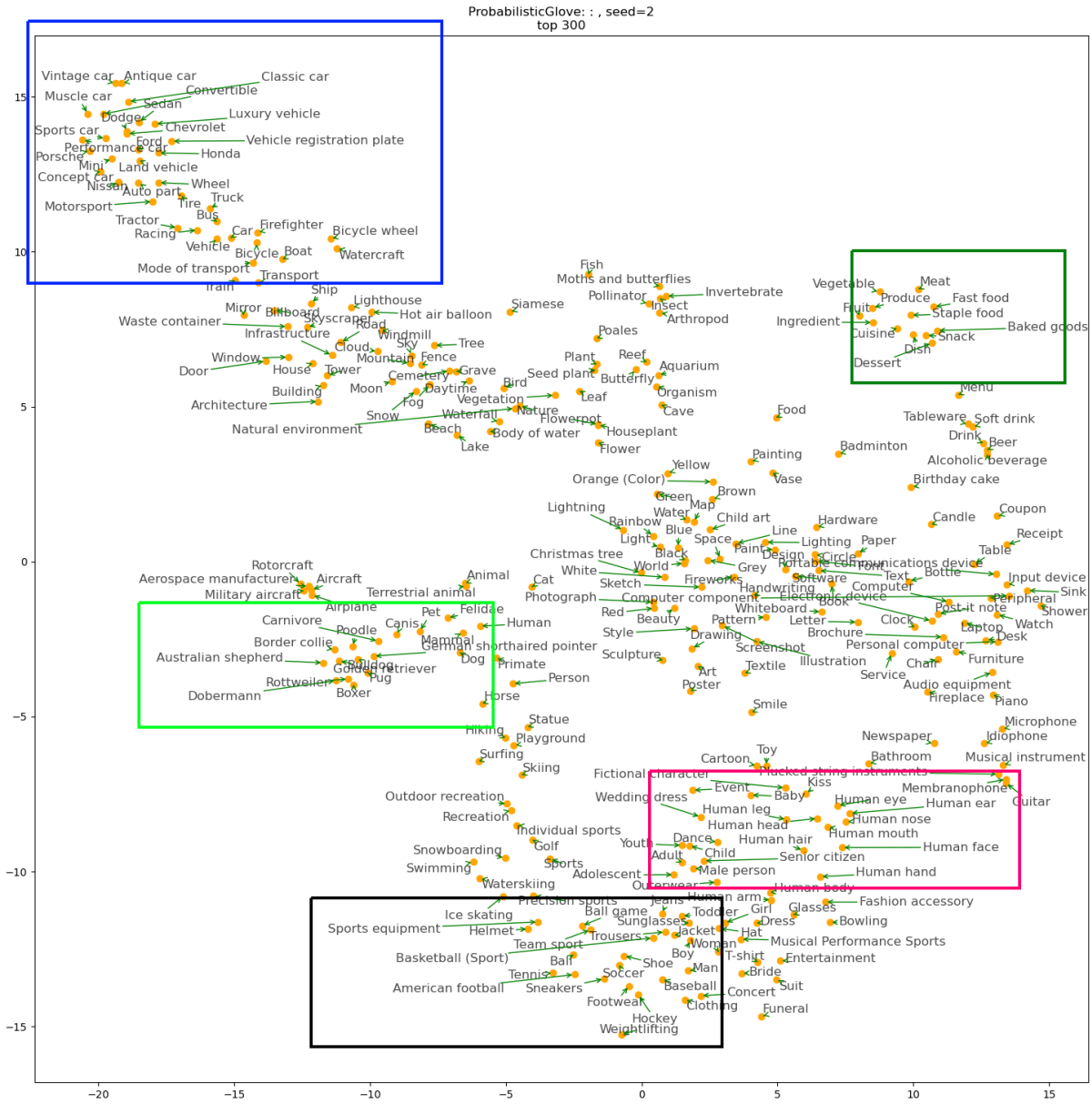

Figure 3.5: The same clusters are in different locations in embedding space compared to the previous run, and their orientation relative to each other is different; it is not a simple rotation and stretch of the clusters. Although t-SNE introduces more stochasticity, both t-SNE runs were done with the same t-SNE random seed set, which is known to generate the same output if given the same inputs.

Learned parameters of the probabilistic embeddings

We also examine the learned means and variances, μ and $\sigma = \ln(1 + \exp(\rho))$, of each embedding.

This is done as follows for each domain (Open Images and AudioSet separately):

- The dot products are calculated between the learned μ (the mean of the stochastic embeddings) for every pair of concepts. The stochastic embeddings are defined as distributions about this mean, where the mean would correspond to the embedding learned in the deterministic case.
- This gives a similarity matrix between each embedding and itself, for each seed.
- The correlation between the values of each run's similarity matrix with every other run was computed, and the mean value taken. A high value should indicate that over runs with different seeds, the similarity of a given pair of concepts is consistent. If a pair is similar in one run, it should be similar in another run and vice versa.
- The average cross-correlation was 0.412 ± 0.045 for AudioSet and 0.590 ± 0.049 for OpenImages.
- The same procedure was done above, but instead of using the learned μ for every pair of concepts, 100 samples of each concept embedding were taken, and 100 similarity matrices calculated using those samples, to get a mean similarity matrix based on sampling. The same cross-correlation was computed, which yielded the values 0.412 ± 0.044 for AudioSet and 0.589 ± 0.049 for OpenImages. This is not a typographical error; the numbers are indeed very similar.

There is a reasonable correlation between the similarity of pairs between runs over both domains.

The variances were examined by analysing the entropy of each embedding. Since each embedding is a multivariate Gaussian with a diagonal covariance matrix:

$$\begin{aligned} H(x) &= \frac{1}{2} \ln |\Sigma| + \frac{D}{2}(1 + \ln(2\pi)) \\ &= \frac{1}{2} \ln \prod_{i=1}^{D} \sigma_i + \frac{D}{2}(1 + \ln(2\pi)) \end{aligned}$$

with σ_i being the variance for dimension i.

Distributional plots of the entropies will give us information about the algorithm's stability over different runs:

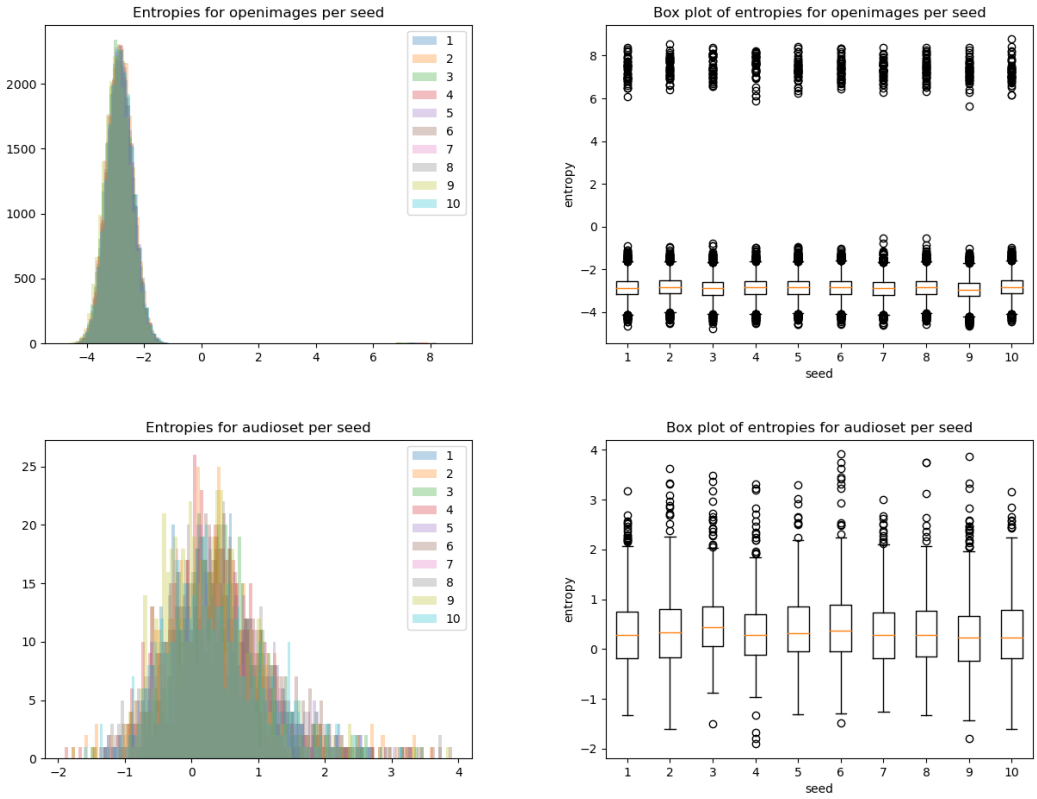

Figure 3.6: Histograms and box plots of entropy distributions for Open Images and AudioSet embeddings.

The plots indicate that the distributions of the variances of learned embeddings are stable over different runs. There is more variability in AudioSet graph as there are only 526 concepts versus 19996 in Open Images. The Open Images distribution is bimodal with a very faint peak centered around 7 on the y-axis.

We expect that concepts that occur fewer times should have a higher variance, and therefore higher entropy. This is confirmed by calculating the Spearman correlation of the entropies with the number of incidences of a concept. For AudioSet, this results in a mean Spearman correlation of -0.35 over 10 random seeds, and for Open Images this is a mean of -0.34 over 10 random seeds.

Unexpected clustering behaviour for some concepts depending on hierarchy

The table below shows the 5 nearest neighbours (in the rows) to the concept "Cat" in the Open Images domain, over 3 seeds. The metric is Euclidean distance. They do not appear to be very related to the concept of "Cat".

Rank / Seed	1	2	3
1	Rope	Cage	Totem pole
	0.290	0.257	0.392
2	Surfing	Human	Sketch
	0.302	0.306	0.422
3	Hammock	Mammal	Peace symbols
	0.302	0.403	0.429
4	Picnic table	Fawn	Beach
	0.364	0.412	0.429
5	Screenshot	Vertebrate	Outdoor furniture
	0.394	0.421	0.438

Table 3.1: The 5 nearest neighbours to the concept "Cat" in the Open Images domain, over 3 seeds. The distance metric is Euclidean distance.

However if we examine the same result for a concept that is a more specific instance of "Cat", we observe much more sensible results for the 5 nearest neighbours. These are the 5 nearest neighbours to the concept "Domestic short-haired cat" in the Open Images domain, also over 3 seeds. Although this is only a qualitative analysis, we observed the same pattern with the nearest neighbours for other concepts in the Open Images appearing in a hierarchy, for example, "Human" compared to "Man" or "Woman", or "Jewellery" compared to "Necklace" or "Ring".

Rank / Seed	1	2	3
1	Malayan cat 0.100	Malayan cat 0.083	Malayan cat 0.144
2	Burmese 0.195	Russian blue 0.118	Himalayan 0.183
3	Russian blue 0.215	Bombay 0.229	Russian blue 0.200
4	Abyssinian 0.228	Polydactyl cat 0.237	Bengal 0.238
5	Tabby cat 0.240	Ragdoll 0.256	Tabby cat 0.267

Table 3.2: The 5 nearest neighbours to the concept "Domestic short-haired cat" in the Open Images domain, over 3 seeds. The distance metric is Euclidean distance.

Additionally, the most specific terms in the domain often have more meaningful nearest neighbours. For example, these are the 5 nearest neighbours to the concept "Roti canai", which is a very specific type of Malaysian flatbread. The nearest neighbours are extremely specific and accurate; "Roti prata", "Paratha", "Naan" and "Uttapam" are all types of flatbread from different Asian countries. We found many other examples of this phenomenon when inspecting the results.

Rank / Seed	1	2	3
1	Roti prata 0.222	Roti prata 0.264	Roti prata 0.145
2	Tortilla de patatas 0.235	Roti 0.299	Pastelón 0.329
3	Pastelón 0.244	Paratha 0.303	Timballo 0.340
4	Paratha 0.287	Uttapam 0.315	Pastitsio 0.350
5	Timballo 0.360	Naan 0.358	Bobotie 0.376

Table 3.3: The 5 nearest neighbours to the concept "Roti canai" in the Open Images domain, over 3 seeds. The distance metric is Euclidean distance.

A possible explanation for this is that embeddings constructed using Euclidean distance measures are known to be less able to represent hierarchical structures, particularly if the embedding space is low-dimensional. [TH86] found that there is quite a restrictive upper bound on the number of points with the same nearest neighbour when the points lie in Euclidean space. When there is a hierarchical semantic structure to the data, nodes that correspond to more abstract types should have many nearest neighbours, because all the child nodes (that are a subtypes of these abstract nodes) should be classed as nearest neighbours. However because of this restrictive upper bound, embeddings learnt in Euclidean space are not able to capture all of these nearest neighbours. [NK17] describes a method of learning embeddings in hyperbolic space (with constant negative curvature) that may be better able to handle this.

The AudioSet nearest neighbours (not shown here for brevity) for concepts were generally more meaningful. This is probably because as there are only 526 concepts represented in AudioSet, there is less of a hierarchy.

3.3 Alignment

3.3.1 Definition of alignment for this problem

As stated in [WPM11], the generic alignment problem involves finding a function that maps one domain to the other. This project involves simultaneously learning this alignment mapping as well as embeddings for both domains. Constraining the GloVe embedding learning problem by inducing alignment should help to decrease the occurrence of local minima and arrive at a better global solution.

We aim for the type of alignment described in Figure 2.4.1, learning a mapping directly from one domain to another, rather than mapping both domains to an intermediate space because at this stage, we cannot make any assumptions about the manifold that the data lie on. We used 6 dimensions based on previous heuristic analysis.

For this problem, the x- and y-embeddings (Open Images and AudioSet respectively) are considered to be aligned if the following holds:

- For every member of the set $x_{intersect} = y_{intersect}$ of concepts that exist in both domains, the nearest neighbour of mapped concept $f(x_i)$ is the corresponding member y_i in set $y_{intersect}$, and the nearest neighbour of mapped concept $g(y_i)$ is the corresponding member x_i in set $x_{intersect}$.

The accuracy is computed after each epoch and defined as the fraction of concepts (in the intersection) in a domain whose nearest neighbour after mapping is the known other embedding in the other domain. Accuracy of 1.0 for domain x denotes that for every concept x_i in domain x, the nearest neighbour of $f(x_i)$ is y_i where it is known that x_i corresponds to y_i.

3.3.2 Alignment model architecture

This project is a proof of concept whose aim is to prove that certain principles are sound, therefore no tuning of architecture or hyperparameters was done.

We implement, in PyTorch, the following alignment network:

- A probabilistic GloVe embedding layer representing the Open Images embeddings to be aligned (henceforth referred to as the x-embeddings).
- A probabilistic GloVe embedding layer representing the AudioSet embeddings to be aligned (henceforth referred to as the y-embeddings).
- A multi-layer perceptron [RHW86] with 3 hidden layers of 100 nodes each and *tanh* activation, that learns a mapping from the x-embeddings to the y-embeddings: $f(x) \to y$.
- A multi-layer perceptron [RHW86] with 3 hidden layers of 100 nodes each and *tanh* activation, that learns a mapping from the y-embeddings to the x-embeddings: $g(y) \to x$.

This particular aligner architecture has been used in previous experiments, with good results.

Full loss function

The full loss function is

$$\begin{aligned} L = & L_{glove,x} + L_{glove,y} + L_{cycle,x} + L_{cycle,y} \\ & + L_{distance,x\text{-}intersect} + L_{distance,y\text{-}intersect} \\ & + L_{distsim,x\text{-}intersect} + L_{distsim,y\text{-}intersect} \end{aligned} \tag{3.3}$$

where the individual components are:

- $L_{glove,x}$: The GloVe loss for x as in 3.2.
- $L_{glove,y}$: The GloVe loss for y as in 3.2.
- $L_{cycle,x}$: The cycle consistency loss from x to y: $||g(f(x)) - x||_2$.
- $L_{cycle,y}$: The cycle consistency loss from y to x: $||f(g(y)) - y||_2$.
- $L_{distance,x\text{-}intersect}$: The distance loss between $f(x)$ and y, for only the intersection of concepts: $||f(x_{intersect}) - y_{intersect}||_2$.
- $L_{distance,y\text{-}intersect}$: The distance loss between $g(y)$ and x, for only the intersection of concepts: $||g(y_{intersect}) - x_{intersect}||_2$.
- $L_{distsim,x\text{-}intersect}$: An optional distributional similarity measure between $f(x_{intersect})$ and $y_{intersect}$.
- $L_{distsim,y\text{-}intersect}$: An optional distributional similarity measure between $g(y_{intersect})$ and $x_{intersect}$.

and $x_{intersect}$ and $y_{intersect}$ represent the embeddings of the concepts that exist in both domains.

Experimental method

For 10 random seeds, one run of the alignment model was done without MMD, then another run with MMD. This results in 20 models, 10 for each random seed without MMD and 10 with MMD.

Other model parameters

Again, because this is a proof of concept, the hyperparameter choices below were not cross-validated or tuned.

- The mini-batch size is 500.
- The Adam [KB17] optimiser is used with a learning rate of 0.01.
- 10 samples of each concept embedding are taken for each mini-batch for a total of 5000 points per mini-batch. As the concept embeddings are themselves distributions, this means that we take 10 samples from each. This acts like data augmentation,

and was necessary to achieve alignment of above 95%. If only one sample was used per concept embedding, maximum alignment accuracy was only around 90%.

- Alignment accuracy is defined as follows: for $x \in x_{intersect}$, the mean number of i such that nearest neighbour of $f(x_i)$ is y_i. For $y \in y_{intersect}$, the mean number of j such that the nearest neighbour of $g(y_j)$ is x_j.
- The model is run for 150 epochs; the mean accuracy (over x and y) is measured after every epoch and the model is saved if the mean accuracy is greater than the last epoch mean accuracy. Therefore, the final saved model has the embeddings that gave the greatest mean accuracy.

In the next sections, the individual items in the loss are discussed more thoroughly.

3.3.3 GloVe loss

The GloVe loss equation from [PSM14] is restated here:

$$L_{glove} = \sum_i \sum_j f(X_{ij})(\boldsymbol{w}_i^T \boldsymbol{w}_j + b_i + b_j - \log X_{ij})^2$$

$$f(x) = \begin{cases} (x/x_{max})^\alpha & \text{if} \quad x \leq x_{max}, \\ 1 & \text{otherwise.} \end{cases}$$

This is a weighted least-squares problem whose solution should be the set of embeddings where the scaled sum of dot products of each pair $\boldsymbol{w}_i$, $\boldsymbol{w}_j$ should approximate the co-occurrence statistics X_{ij}.

Experimentally, it was found that the GloVe loss for the respective domains had to be scaled by the ratio of concepts: there are 19996 Open Images concepts and 526 Audioset concepts, so the x glove loss (Open Images) is multiplied by 19996 / 526. Without this rescaling, the learned embeddings yielded non-sensible results (see Figure 4.7).

3.3.4 Cycle consistency

The cycle-consistency loss relates a domain's embeddings to their reconstruction after mapping to the other domain and back. This uses transitivity to self-supervise training by inducing $g(f(x))$ to be close to x and $f(g(y))$ to be close to y.

As stated in [Zhu+17], simply requiring $f(x)$ to be close to y and $g(y)$ to be close to x is insufficient as overfitting can result in a network being able to map an arbitrary set of inputs to an arbitrary set of outputs. In the CycleGAN implementation of [Zhu+17], the L1 norm is used, but in this experiment we use the L2 norm as we do not want a sparsity constraint imposed on the embeddings.

Thus the cycle consistency loss takes the following form:

$$L_{cycle} = E_x||f(g(x)) - x||_2 + E_y||g(f(y) - y||_2$$

This loss is also used in other alignment architectures, for example in the MAGAN, [AK18] where it is called reconstruction loss.

3.3.5 Distance loss

This measures how far away the mapped embedding is from the original embedding. This loss represents the semi-supervised component of the algorithm; it is only calculated for the 230 concepts present in both Open Images and AudioSet.

$$L_{distance} = E_{x_{intersect}}||f(x_{intersect}) - y_{intersect}||_2 + E_{y_{intersect}}||g(y_{intersect}) - x_{intersect}||_2$$

The Manifold Alignment GAN (MAGAN [AK18]) used this loss to prevent the generators (the mapping MLPs in our model) learning a mapping that simply superimposed the manifolds without actually aligning the points. In our model it has the same function- this is the loss that induces the mapping from one domain to another.

3.3.6 Distributional similarity measure

We want the distributions of $f(x_{intersect})$ and $y_{intersect}$ to be similar, and likewise for $g(y_{intersect})$ and $x_{intersect}$. To be usable as loss functions, a distributional similarity measure should generate a scalar value that is minimised when the distributions are identical.

Maximum mean discrepancy

The maximum mean discrepancy (MMD) statistic is introduced in [Gre+12] as a difference in feature means of two distributions, where features are obtained by applying a kernel function to the two samples. If the kernel function meets certain conditions, the MMD between samples of two distributions is zero if the distributions are identical. Summarising the full theory that can be found in [Gre+12], the following conditions must hold:

- The kernel must be characteristic: If the inputs are random variables X with domain Ω and distribution P, there is a one to one mapping between the mean value in feature space and the distributions. [Mua+17]
- That is, each distinct value for the mean in feature space maps to a different possible distribution.
- Intuitively, the kernel function must be "rich enough" to represent all possible distributions. A known characteristic kernel function that is commonly used is the Gaussian kernel function $k(\boldsymbol{x}, \boldsymbol{y}) = \exp(-\alpha||\boldsymbol{x} - \boldsymbol{y}||_2^2)$, and this is what we use in our experiments.

Given the following:

- $\boldsymbol{x}_i$ are m samples from one distribution X
- $\boldsymbol{y}_j$ are n samples from the other distribution Y
- $k(\boldsymbol{x}, \boldsymbol{x}')$ is a characteristic kernel function

The empirical unbiased MMD statistic is as follows:

$$\begin{aligned} MMD(k, \boldsymbol{x}, \boldsymbol{y}) = &\frac{1}{m(m-1)} \sum_{i=1}^{m} \sum_{j \neq i}^{m} k(\boldsymbol{x}_i, \boldsymbol{x}_j) + \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j \neq i}^{n} k(\boldsymbol{y}_i, \boldsymbol{y}_j) \\ &- \frac{2}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} k(\boldsymbol{x}_i, \boldsymbol{y}_j) \end{aligned}$$

This is the sum of the within-distribution similarities, less the sum of the cross-distributional similarities.

The implementation from the `torch-two-sample` [1] library [DK17] was used.

α of 0.01 was used for both domains. This value was chosen because it is close to $\frac{1}{2\sigma^2}$ where σ is the median of the pairwise distances, which is a good heuristic for convergence [GJK17] .

[1] `https://torch-two-sample.readthedocs.io`

Chapter 4

Results

In this chapter we present the results of semi-supervised learning of jointly aligned embeddings, where "jointly aligned" means that embeddings for both domains and the mapping from one to the other are all learned simultaneously.

4.1 Summary

- Alignment accuracies of greater than 95% were achieved for both domains ($f(x) \rightarrow y$ and $g(y) \rightarrow x$). If the domain embeddings are aligned, it means that our multi-task learning problem is well constrained.
- Visualisation of the jointly aligned embeddings through t-SNE dimensionality reduction showed good clustering behaviour with semantically meaningful clusters.
- **Aligned embeddings were found to be of higher quality compared to independently learned embeddings, when using a similarity metric based on the WordNet lexical database.** The specific definitions of similarity and quality are described in Sections 4.3.1 and 4.4.
- Aligned embeddings were found to be less stable compared to independently learned embeddings. The definition of stability is described in Section 4.4.4.
- The MMD metric increased alignment accuracy, but sometimes decreased embedding quality.

4.2 Visualising clusters

Similar to what was done with the independently learned embeddings, we run the t-SNE algorithm on the aligned embeddings for both domains, and plot the results.

In Figures 4.1 and 4.2 below, the blue points represent the concepts in the intersection i.e. they are present in both Open Images and AudioSet. The orange points represent the concepts that are in the 300 most frequent for the respective domain that are not also in the intersection. Only the t-SNE plots for alignment run without MMD are shown, as the plots for alignment run with MMD look materially similar.

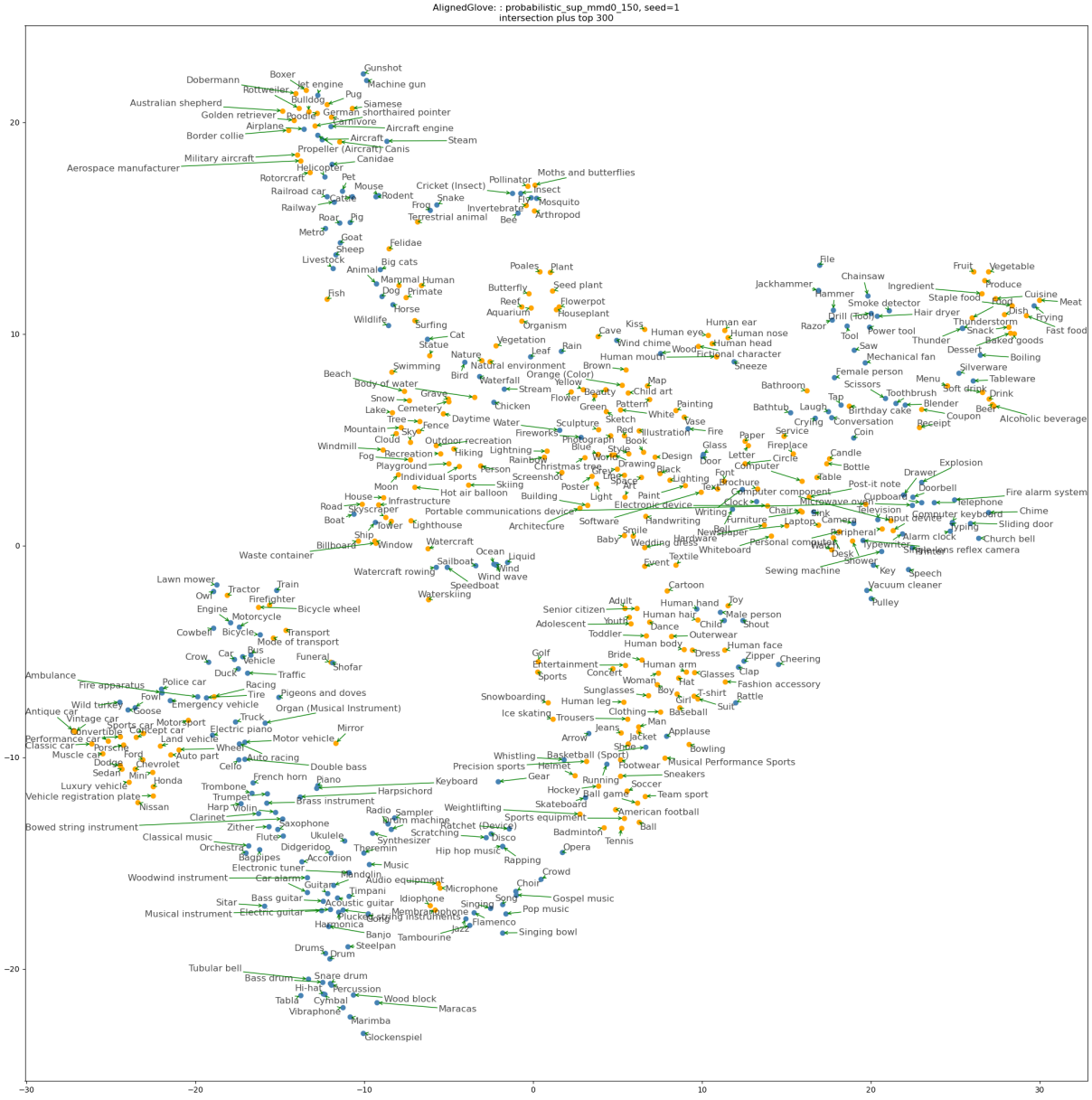

Figure 4.1: t-SNE plot of concept embeddings for Open Images that are in the intersection of both domains (blue points) or in the top 300 most frequent (orange points). These are the embeddings for alignment run without MMD. See Figure 4.3 for zoomed-in examples.

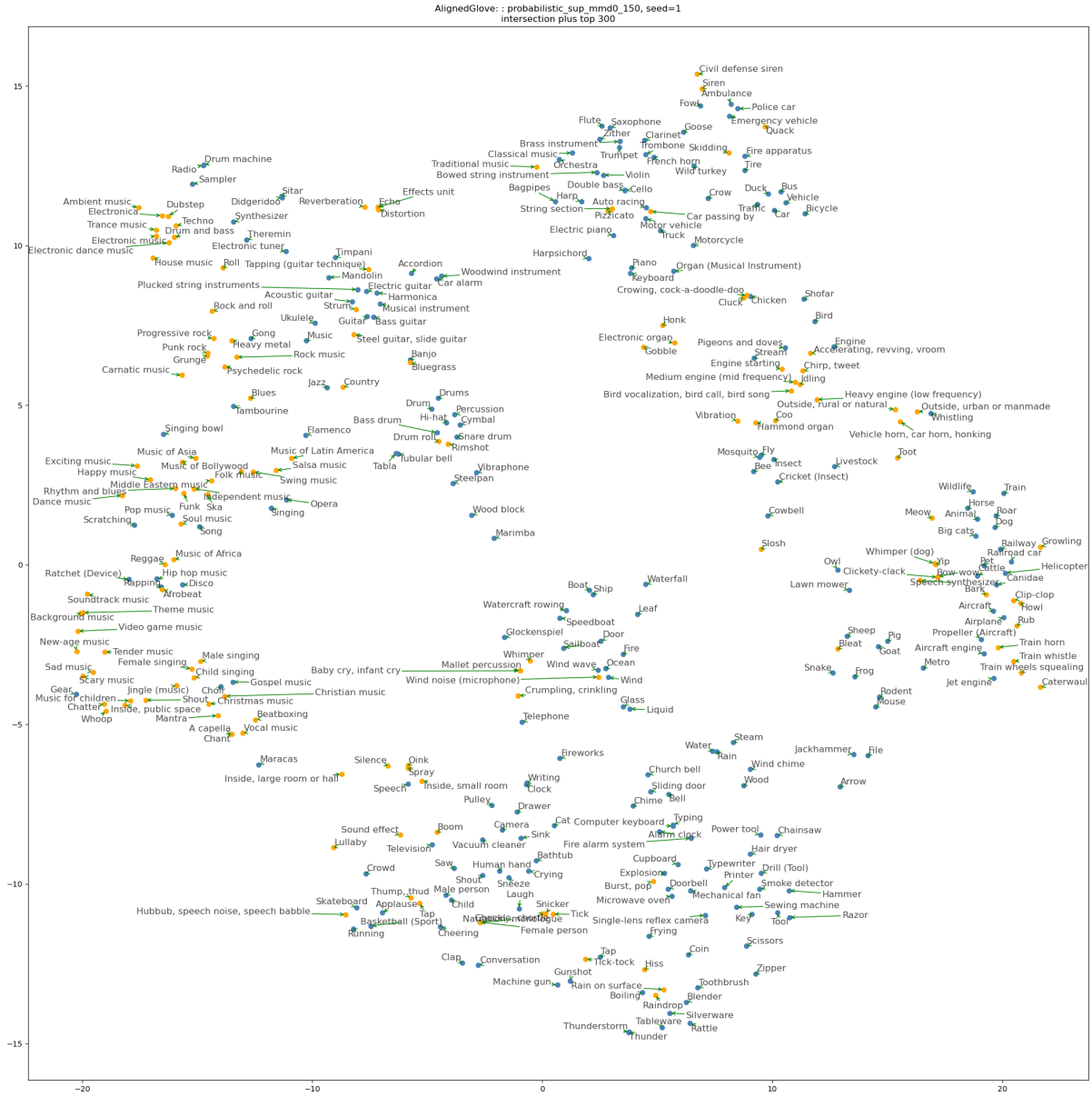

Figure 4.2: t-SNE plot of concept embeddings for AudioSet that are in the intersection of both domains (blue points) or in the top 300 most frequent (orange points). These are the embeddings for alignment run without MMD. See Figure 4.4 for zoomed-in examples.

In Figures 4.3 and 4.4 below, we show some enlarged examples of clusters which show concepts in the intersection and out of the intersection. These clusters show that a good balance is struck between the GloVe loss and the distance loss / MMD; for an example of what happens when this balance is not found, see Figure 4.7, where the GloVe loss was

not weighted sufficiently compared to the distance loss / MMD.

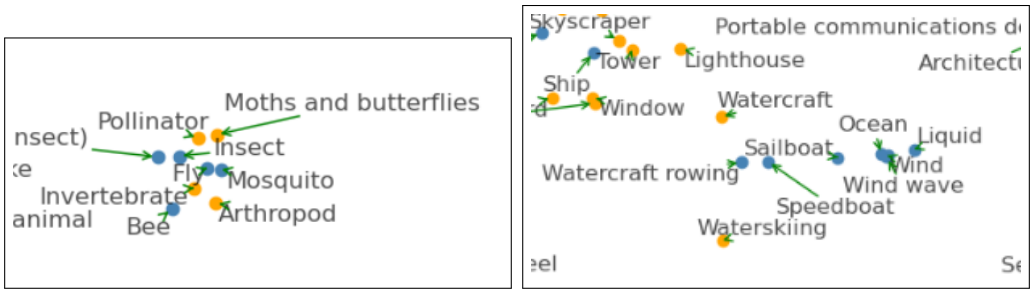

Figure 4.3: Enlarged view of two regions of the Open Images t-SNE plot, showing that concepts in and out of the intersection but semantically similar do form good clusters. As blue points denote concepts present in both domains and orange points denote concepts amongst the most frequent in the domain being examined, we want a mixture of orange and blue.

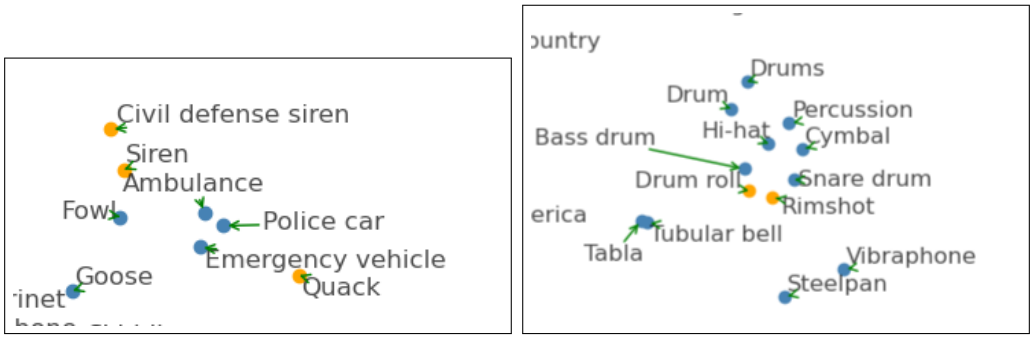

Figure 4.4: Enlarged view of two regions of the AudioSet t-SNE plot showing the same phenomenon as described above.

4.3 Statistics

In this and following sections, the notation 1.234 ± 0.56 for an item denotes that the mean of N (usually 10) samples of the items' observations is 1.234 and the standard deviation is 0.56.

4.3.1 Similarity and correlations

We examine the correlations of self-similarity matrices (as described in Section 3.2.2) over different random seeds, to get an idea of how consistent each set of embeddings is with itself. The similarity matrix of the embeddings for a given run is like a "signature" of how the different concepts relate to each other within that run. We would like a high correlation of similarity matrices between runs, because that tells us that the algorithm is producing stable embeddings where concepts have the same degree of similarity with other concepts over many runs.

Pearson cross-correlation of self-similarity matrices of embedding means

	Independent	Aligned without MMD	Aligned with MMD
Open Images	0.590 ± 0.049	0.474 ± 0.044	0.420 ± 0.040
AudioSet	0.412 ± 0.045	0.399 ± 0.044	0.346 ± 0.043

Table 4.1: The mean and standard deviation of Pearson correlation of self-similarity matrices with themselves, taken over 10 random seeds. The similarity measure used for this table is the dot product of the means of the embeddings (μ in Equation 3.1). The mean is taken over all combinations of random seeds (1x2, 1x3, ..., 9x10).

Pearson cross-correlation of self-similarity matrices of samples of embeddings

	Independent	Aligned without MMD	Aligned with MMD
Open Images	0.589 ± 0.049	0.480 ± 0.040	0.434 ± 0.040
AudioSet	0.412 ± 0.044	0.399 ± 0.044	0.346 ± 0.043

Table 4.2: The mean and standard deviation of Pearson correlation of self-similarity matrices with themselves, taken over 10 random seeds. The similarity measure used for this table is the mean of 100 sampled dot product similarity matrices (100 samples are taken, the pairwise dot product similarity computed, and the mean of those 100 similarity matrices is taken). This measure was used to take into account the variance of the embeddings, where the previous table used only the learned means of the embeddings. The average is taken over all combinations of random seeds (1x2, 1x3, ..., 9x10). It is not a typographical error that the values for AudioSet for mean of 100 similarity matrices are the same as the values for the embedding means in the previous table.

The runs of the independently learned embeddings are more correlated with each other, compared to the equivalent calculation with the aligned embeddings. This means that the similarity matrices (of each run's embeddings with itself) are more consistent across runs. This correlation has dropped in the aligned runs, indicating that the output embeddings are less similar to themselves across runs. We can say that the alignment introduces more variation in the embeddings. MMD in particular seems to introduce more variability.

Spearman correlation of entropy with frequency of occurrence

For the independently learned embeddings, we observe a slight negative Spearman correlation of the learned entropy of a concept's embeddings with the frequency of occurrence of that concept. Entropies of less frequently occurring concepts are found to be higher, which is intuitively sensible; there is less information about those concepts so we would expect the variance, and therefore the entropy, to be higher.

	Independent	Aligned without MMD	Aligned with MMD
Open Images	-0.179 ± 0.011	0.0421 ± 0.019	0.0362 ± 0.012
AudioSet	-0.400 ± 0.041	-0.000539 ± 0.074	0.0359 ± 0.026

Table 4.3: The mean and standard deviation of Spearman correlation of entropy with frequency of occurrence, taken over 10 random seeds.

This correlation has largely disappeared in the aligned cases, being roughly around zero, indicating that there is now no relationship between the variance of the learned embedding for a concept and the frequency of that concept in the dataset. In this respect, the variances of the aligned probabilistic embeddings are less informative than the variances of the independent probabilistic embeddings.

4.3.2 Entropy plots of aligned embeddings

Open Images

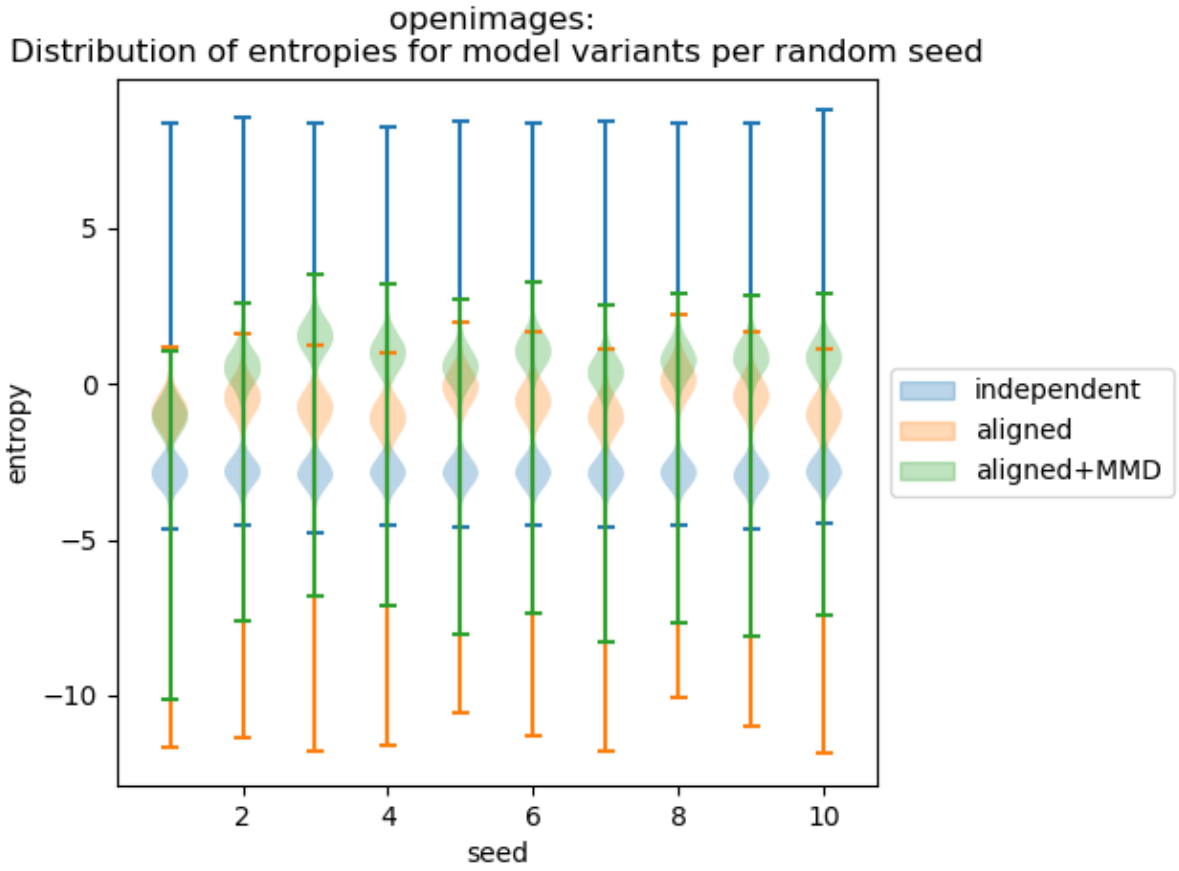

Figure 4.5: Violin plots of entropy distributions for independent embeddings, aligned without MMD and aligned with MMD. The entropy distribution for aligned embeddings is more variable between runs than that for independent embeddings. The use of MMD increases this variability. These results are consistent with the self-similarity correlations between runs shown in Tables 4.1 and 4.2, which also indicated that the aligned embeddings were more unstable over different runs.

AudioSet

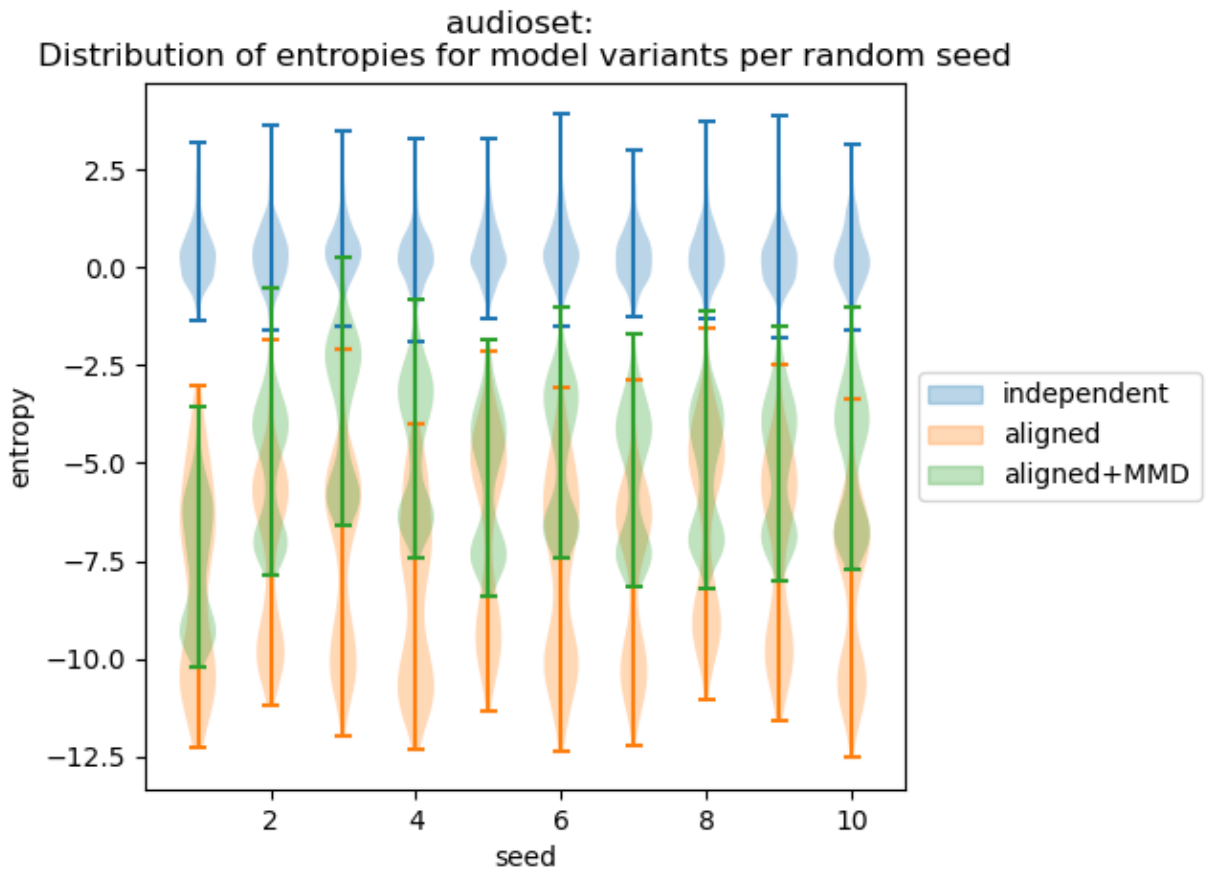

Figure 4.6: Violin plots of entropy distributions for independent embeddings, aligned without MMD and aligned with MMD. The aligned embeddings, whether run with or without MMD, show a bimodal entropy distribution and much greater variability than that of the independent embeddings. These results are consistent with the self-similarity correlations between runs shown in Tables 4.1 and 4.2, which also indicated that the aligned embeddings were more unstable over different runs.

4.4 Alignment accuracy and embedding quality

The table below shows the alignment accuracy for both domains, over 10 random seeds. Using the MMD statistic as a component of the loss marginally increased the accuracy.

	Open Images		Audioset	
Seed	Without MMD	With MMD	Without MMD	With MMD
1	0.9479	0.9478	0.9478	0.9652
2	0.9565	0.9826	0.9696	0.9826
3	0.9348	0.9565	0.9565	0.9783
4	0.9522	0.9652	0.9565	0.9565
5	0.9478	0.9609	0.9435	0.9696
6	0.9522	0.9739	0.9739	0.9696
7	0.9652	0.9652	0.9478	0.9739
8	0.9696	0.9522	0.9609	0.9522
9	0.9565	0.9565	0.9609	0.9783
10	0.9609	0.9522	0.9652	0.9783
mean	0.9543	0.9613	0.9583	0.9704

Table 4.4: The alignment accuracy for each run of the models for both domains, with and without MMD.

High alignment accuracy between embeddings in both domains does not necessarily mean the embeddings are good. Figure 4.7 below is a t-SNE plot of aligned embeddings (reduced to 2 dimensions) of very high accuracy (97%), meaning that the embeddings in Open Images and AudioSet are well aligned. However, the actual embeddings did not form good clusters; concepts that were related semantically tended not to be close in embedding space.

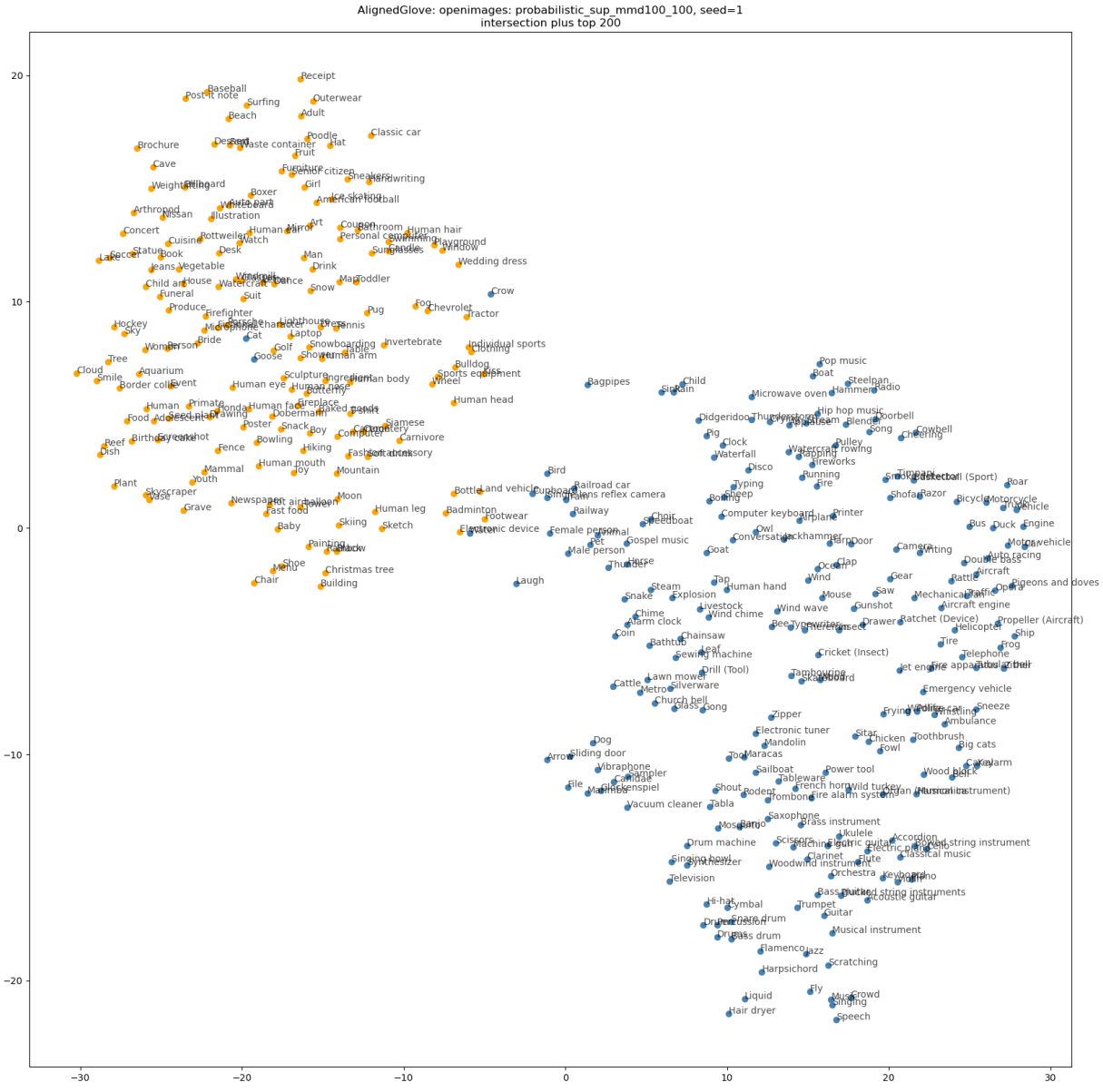

Figure 4.7: The alignment accuracy for this set of embeddings is 97%. The blue points denote concepts that are in the intersection of concepts (present in both Open Images and AudioSet). The orange points denote concepts that are in the 200 most frequent concepts occurring in Open Images. It is immediately visible that the algorithm has clustered concepts in the intersection degenerately; concepts that are in the intersection are more likely to be close to other concepts in the intersection rather than concepts that are semantically close.

Hence, alignment accuracy is insufficient as a metric. We can also examine the average Pearson correlations of the similarity matrices of the aligned and independent embeddings over 10 random seeds, for the two model variants:

Open Images		Audioset	
Without MMD	With MMD	Without MMD	With MMD
0.532 ±0.035	0.465 ±0.038	0.334 ±0.036	0.331 ±0.038

Table 4.5: The mean and standard deviation of the Pearson correlation of the self-similarity matrices of aligned and independent matrices, over 10 random seeds.

The above are aggregate statistics that only tell us that there is a positive correlation between the similarities of the aligned and independent embeddings. We do not know if the aligned embeddings are actually semantically more meaningful than the independent, or the other way around.

4.4.1 Comparing aligned embeddings with human similarity scores

A plausible measure of embedding quality would be a comparison of the similarity of concept pairs as calculated from embeddings, with the similarity of concept pairs as evaluated by humans. In order to do this, we need to find datasets that capture human ratings of similarity. If our alignment algorithm is good, we should expect that the aligned embeddings are of higher quality than the independently learned embeddings and correlate more highly with human similarity judgement.

MTURK-771: The MTURK-771 dataset was created by the authors of [Hal+12], a study in learning the relatedness of word pairs. The predictions are tested by comparing the Spearman correlation of its predictions with human judgements. The MTURK-771 dataset comprises 771 pairs of words with human-rated similarity scores, collected using the Amazon Mechanical Turk tool. The intersection of the 771 word pairs with our Open Images and AudioSet concepts was small - 171 (0.013%) for Open Images and only 3 (0.0071%) for AudioSet.

WordNet: The WordNet [Mil95] lexical database contains English nouns, verbs and adjectives grouped according to synonymy (semantic similarity) and hypernymy/hyponymy (hierarchy). Words are represented by "synsets" which denote a particular sense of the word (for example, "orange" may refer to the colour or the fruit). Therefore, one word may map

to several synsets. The relationships between word senses were hand-encoded from various corpora and thesauri, which themselves were human-curated; therefore, we consider the lexical and semantic information contained within WordNet to be an appropriate representation of human judgement. We accessed the WordNet database through the `nltk` Python package.

There is a substantial overlap between pairs of words found in WordNet and pairs of concepts from our dataset. Out of approximately 128 million (128005400) nonzero pairs occurring in Open Images, approximately 27 million (27052350, 21.1%) are also present in WordNet. Out of 42002 nonzero pairs in AudioSet, 19208 (45.7%) are also present in WordNet.

ILSVRC: A third available dataset originates from a prototype model for [RL20a], in which human similarity judgements of pairs of concepts were collected to supplement the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) 2012 task. This dataset will be referred to as the "enhanced ILSVRC" dataset. These judgements were collected from participants using Amazon Mechanical Turk. The data collection process used probabilistic techniques for trial selection to maximise the expected information gain from each trial. In this dataset, 147070 (0.115%) pairs overlap with Open Images, and 93 (0.221%) pairs overlap with AudioSet. One word, "tick", was removed from the AudioSet pairs when computing the comparison, because that label as used in AudioSet is used to describe the sound and not the insect. We know that the ImageNet label corresponds to the insect.

We adopt the method used by [Hal+12] of comparing the Spearman correlation of the similarity of our embedding concept pairs with the similarity of the human-measured dataset to evaluate the quality of aligned embeddings compared to independently learned embeddings. The Spearman correlation is used because we wish to evaluate whether there is a monotonic relationship, and the similarity measures all use different scales. The cosine similarity measure is used as it contains the dot product, which is an input into the GloVe algorithm that generates the embeddings, and the magnitude of the embeddings is not meaningful.

The similarity of two concepts with indexes i and j in embedding space is:

$$s_{ij} = \frac{\boldsymbol{e}_i \cdot \boldsymbol{e}_j}{||\boldsymbol{e}_i||_2 \, ||\boldsymbol{e}_j||_2}$$

where $\boldsymbol{e}_i$ represents the i-th embedding.

If the correlation with the human similarity dataset is higher for aligned embeddings than for independently learned embeddings, this indicates that

the alignment process is adding value, resulting in more cognitively plausible embeddings. This is because it indicates that the aligned embedding pairs are in aggregate more like human judgement of similarity than the independent embedding pairs. We will use this as a metric for embedding quality, where higher quality means more similar to human judgement. We evaluate the model variants with and without MMD using this metric to assess the effect of MMD.

The MTURK-771 and enhanced ILSVRC datasets are pre-existing mappings of concept pairs to similarity values, which can be used without any preprocessing. There is no such WordNet dataset of direct similarity, so we have to construct one. WordNet makes available similarity measures between synsets that can be computed directly from the Python library. We choose the Leacock-Chodorow (LCH) similarity measure [LMC98], which takes the shortest path between the two synsets, scaled by the maximum depth from the top of the taxonomy tree of the two synsets (this measure was chosen as it is the only one that combines path length between nodes and depth of the tree).

As a single word or phrase may map to several synsets, and we have no sense information in our Open Images / AudioSet concept names to disambiguate between choices, we may have more than one feasible synset pair per concept pair in our dataset. We use the following heuristic algorithm to decide which specific synset pair's scores to use:

- Check every pair in the domain (Open Images or AudioSet) that appears at least once (its entry in the co-occurrence matrix is nonzero).
- Convert the pair words to lowercase with spaces replaced with underscores (this is the WordNet naming scheme).
- Check if both words have synsets in WordNet. If either does not, ignore the pair.
- For each combination of the first 2 synsets for each word (up to 4 pairs in total), compute the LCH similarity if both elements in the pair have the same part of speech, otherwise ignore the pair. [1] This is to handle pairs like "mandarin orange" and "orange". "Mandarin orange" has two synsets; the first refers to the mandarin orange tree, and the second refers to the fruit. Therefore the second "mandarin orange" synset matches the first "orange" synset (which denotes the fruit, rather than the colour) more closely. One mode of failure for this algorithm is for words that occur in both Open Images and AudioSet but with different meanings, for example, "tap" and "tick" are different concepts when referring to objects or sounds.

[1] WordNet similarity is not defined for synsets which do not have the same parts of speech.

- The WordNet similarity for the pair is taken to be the largest such value. This algorithm will therefore be biased high.

4.4.2 Results of comparison with human similarity metrics

Degree of overlap of human similarity datasets with domain pairs

These are the percentages of data overlap for the different domains. WordNet is the dataset with the largest overlap.

Human similarity dataset	Open Images	AudioSet
MTURK-771	0.013%	0.0071%
WordNet	21.1%	45.7%
enhanced ILSVRC	0.115%	0.221%

Table 4.6: The proportion of pairs of concepts that overlap between domains (in columns) and human similarity datasets (in rows).

Open Images, Spearman correlation with human similarity metrics

	MTURK-771			WordNet			ILSVRC		
Seed	Ind.	Aligned	Aligned +MMD	Ind.	Aligned	Aligned +MMD	Ind.	Aligned	Aligned +MMD
1	0.357	0.306	0.331	0.205	0.229	0.228	0.524	0.493	0.488
2	0.343	0.271	0.296	0.196	0.221	0.230	0.471	0.460	0.458
3	0.376	0.338	0.287	0.200	0.240	0.240	0.483	0.486	0.433
4	0.343	0.300	0.290	0.210	0.233	0.229	0.514	0.531	0.482
5	0.358	0.256	0.225	0.191	0.226	0.226	0.505	0.466	0.456
6	0.374	0.309	0.271	0.207	0.239	0.237	0.541	0.473	0.451
7	0.347	0.297	0.292	0.208	0.229	0.239	0.506	0.468	0.447
8	0.343	0.284	0.283	0.215	0.217	0.218	0.553	0.458	0.468
9	0.340	0.279	0.271	0.217	0.224	0.233	0.508	0.463	0.439
10	0.367	0.267	0.284	0.207	0.225	0.230	0.531	0.464	0.452
mean	0.355	0.291	0.283	0.205	0.228	0.231	0.514	0.476	0.458

Table 4.7: The mean Spearman correlation of the human-judged similarity measure, and cosine similarity of pairs between embeddings (for different model variants: independent, aligned without MMD and aligned with MMD). Results are shown for each random seed. The mean for each seed is taken over all pairs of concepts present in both the embeddings and the human similarity dataset. The mean correlation (bottom row) is taken over all seeds.

To present a different view, we use the Spearman correlation of the independently aligned embedding pair similarity and the human similarity dataset as a baseline. We then subtract this from the model variant embedding pair similarity correlation, to get an idea if the model variant adds value by increasing the correlation (making the resulting embeddings more similar with human judgement than the independently learned embeddings).

	MTURK-771		WordNet		ILSVRC	
Seed	Aligned	Aligned +MMD	Aligned	Aligned +MMD	Aligned	Aligned +MMD
1	-0.0512	-0.0260	0.0244	0.0234	-0.0305	-0.0353
2	-0.0716	-0.0465	0.0255	0.0346	-0.0107	-0.0130
3	-0.0374	-0.0884	0.0402	0.0402	0.00349	-0.0495
4	-0.0433	-0.0530	0.0232	0.0185	0.0174	-0.0322
5	-0.102	-0.133	0.0354	0.0356	-0.0385	-0.0484
6	-0.0650	-0.103	0.0320	0.0301	-0.0686	-0.0904
7	-0.0509	-0.0552	0.0216	0.0308	-0.0383	-0.0595
8	-0.0592	-0.0599	0.00189	0.00313	-0.0948	-0.0851
9	-0.0614	-0.0686	0.00696	0.0166	-0.0448	-0.0691
10	-0.0995	-0.0828	0.0184	0.0235	-0.0674	-0.0793
mean	-0.0641	-0.0716	0.0230	0.0256	-0.0373	-0.0562

Table 4.8: The differences between Spearman correlations of similarity of aligned model variants with human similarity, and the equivalent for independently learned embeddings.

For Open Images, using the WordNet comparison metric, the aligned embeddings are more correlated with human similarity than the independently learned embeddings, as the mean Spearman correlation of embedding pairwise cosine similarity with WordNet similarity is greater for aligned embeddings than for independent embeddings.

The reverse is observed with the MTURK-771 comparison metric, but we do note that this is a very small dataset of only 170 pairs present in both Open Images (out of 120 million pairs) and MTURK-771. The same phenomenon as with MTURK-771 is also observed when comparing with the ILSVRC metric, where the aligned embeddings are less correlated with human judgement than the independently learned embeddings' correlation with human judgement.

However, the ILSVRC dataset is unbalanced, and some of the choices of included concepts are strange. There are 1000 concepts present in it, of which 124 are different breeds of dog. In fact, 398 of the concepts present in the ILSVRC are different types of animal. There are some concepts in the selected 1000 that almost certainly are not amongst the most common humanly known concepts, for example, "shoji"[2] and "bicycle-built-for-

[2]A Japanese sliding door made of paper.

two" (with "bicycle" not present). The only flowers present are "cardoon" [3], "daisy" and "yellow lady's slipper". In short, the concepts represented in the ILSVRC dataset do not appear to be a very good sample of human concepts. To investigate whether the imbalance of the dataset was affecting results, runs were tried of the ILSVRC dataset excluding all the animals and then only including animals, but the results were broadly the same.

Given that the overlap of WordNet pairs with our domain pairs is considerable (21.1% for Open Images and 45.7% for AudioSet), we think that the higher correlation with WordNet similarity for aligned Open Images embeddings constitutes evidence that embedding quality is improved by alignment. This would be consistent with hypotheses that multi-task learning adds value by producing a model that generalises better. While the AudioSet concept universe is not large, there are still 200+ concepts that are not present in Open Images that provide the GloVe algorithm with more information than would otherwise be present in the independently learned case.

However, playing devil's advocate, we do also point out that the WordNet similarity score is an artificial score inferred from properties of the WordNet database, whereas the MTURK-771 and ILSVRC similarity measures are directly collected from humans.

Using the MMD as a component of the loss appears to increase accuracy, as well as alignment quality as measured against the WordNet metric. If comparing to the MTURK-771 and ILSVRC datasets, using MMD appears to decrease the alignment quality.

[3] A type of thistle.

AudioSet, Spearman correlation with human similarity metrics

MTURK-771 is excluded from this comparison, as with only 3 pairs present in both MTURK-771 and AudioSet, no meaningful results could be obtained.

	WordNet			ILSVRC		
Seed	Ind.	Aligned	Aligned +MMD	Ind.	Aligned	Aligned +MMD
1	0.139	0.163	0.153	0.635	0.683	0.502
2	0.132	0.130	0.147	0.585	0.687	0.651
3	0.157	0.175	0.133	0.647	0.428	0.617
4	0.147	0.148	0.169	0.560	0.610	0.649
5	0.116	0.151	0.152	0.562	0.513	0.578
6	0.144	0.134	0.125	0.631	0.609	0.504
7	0.150	0.134	0.170	0.539	0.563	0.546
8	0.140	0.183	0.128	0.553	0.675	0.682
9	0.147	0.136	0.187	0.524	0.718	0.681
10	0.150	0.166	0.143	0.607	0.717	0.699
mean	0.142	0.152	0.151	0.584	0.620	0.611

Table 4.9: The mean Spearman correlation of the human-judged similarity measure, and cosine similarity of pairs between embeddings (for different model variants: independent, aligned without MMD and aligned with MMD). Results are shown for each random seed. The mean for each seed is taken over all pairs of concepts present in both the embeddings and the human similarity dataset. The mean correlation (bottom row) is taken over all seeds.

Analogous to what is done with Open Images, we use the Spearman correlation of the independently aligned embedding pair similarity and the human similarity dataset as a baseline, to evaluate if the model variant adds value by making the resulting embeddings more similar with human judgement than the independently learned embeddings.

	WordNet		ILSVRC	
Seed	Aligned	Aligned +MMD	Aligned	Aligned +MMD
1	0.0237	0.0138	0.0489	-0.132
2	-0.00278	0.0144	0.102	0.0662
3	0.0176	-0.0237	-0.219	-0.0293
4	0.00162	0.0221	0.0497	0.0886
5	0.0350	0.0361	-0.0488	0.0160
6	-0.00988	-0.0189	-0.0218	-0.127
7	-0.0156	0.0202	0.0237	0.00677
8	0.0423	-0.0127	0.122	0.129
9	-0.0117	0.0401	0.194	0.157
10	0.0160	-0.00696	0.110	0.0918
mean	0.00964	0.00844	0.0361	0.0267

Table 4.10: The differences between Spearman correlations of similarity of aligned model variants with human similarity, and the equivalent for independently learned embeddings.

When compared with both WordNet and ILSVRC, AudioSet aligned embeddings are more correlated with human similarity judgement than the independently learned embeddings. However, only 93 pairs present in AudioSet overlap with ILSVRC pairs, so this is not a large sample for comparison. It is also subject to the vagaries of the ILSVRC dataset concept choice, as discussed in the previous section. We observe an asymmetry, where Open Images improves AudioSet with respect to ILSVRC, but AudioSet does not improve Open Images. This could be because there are nearly 40 times as many concepts in Open Images than AudioSet, so there is simply much more information there to assist with alignment of AudioSet than there is in the other direction.

Similar to Open Images, including MMD in the loss function increases accuracy, but it slightly decreases alignment quality when compared to independently learned embeddings using both WordNet and ILSVRC metrics.

4.4.3 Overall results of similarity comparison

When compared with WordNet similarity, both domains' aligned embeddings showed more correlation with human similarity judgement than the equivalent independently learned embeddings. This is some evidence to show that the information gained from learning both domains simultaneously results in both domains' embeddings being of higher quality. In particular, the embedding quality was improved for the smaller domain AudioSet.

This is consistent with our hypothesis that constraining the embedding learning problem for both domains by adding alignment would allow each domain to act as an inductive bias for the other, leading to a better global solution. This also matches the multi-task learning research showing that learning from multiple sources results in models that generalise better.

4.4.4 Embedding stability

Another metric of embedding quality is their stability over multiple runs. As previously mentioned, the stochasticity in the GloVe embedding training means that different embeddings are produced for different runs, corresponding to different local minima. The structures of these are not the same from run to run, as we saw in a table of the top 5 nearest neighbours for various concepts, over different random seeds (Tables 3.1, 3.2).

We use two measures of stability; one is the metric mentioned in [BKM20], a simple count of the number of intersecting concepts in the 5 nearest neighbours of any concept, over 10 seeds (divided by 5 to normalise to 1). We also define an alternate similarity metric which is easier to plot and visualise as follows:

- Measure the size of the union of all the top 5 nearest neighbours of a concept, by Euclidean distance, for 10 random seeds. Let this be C.
- Take the mean of $5/C$ for the top 300 concepts in Open Images and Audioset.
- If the embeddings are very stable, then the top 5 nearest neighbours of a concept over all runs should be the same, and the stability will be 1. If they are very unstable, the top 5 nearest neighbours will be different with each run, and the stability will be a small number.

Mean stability of embeddings by model variant

Empirically, alignment appears to decrease stability of the embeddings, meaning that the nearest neighbours of a particular concept (measured by Euclidean distance) tend to change more over different random seeds. Intuitively, this is reasonable. Training aligned probabilistic embeddings involves more variables and a more complex loss function, so we could expect the variances to be greater.

Embedding type	MMD	Domain	Union metric	Intersection metric
Aligned	Without MMD	Open Images	0.2373	0.0733
		AudioSet	0.2108	0.0535
	With MMD	Open Images	0.2103	0.0527
		AudioSet	0.2248	0.0816
Independent	N/A	Open Images	0.2900	0.1013
		AudioSet	0.3901	0.3311

Table 4.11: The mean stability by both metrics for all model variants, over both domains. The stability is measured for the top 300 most frequent concepts in each domain.

Plots of stability by relative frequency

The per-item plots show very clearly that alignment decreases the stability of the embeddings, as seen from the distribution of the stability values per rank of concept frequency. For the independently learned embeddings, some concepts have a stability of 1.0 which means that over 10 random seeds, the 5 nearest neighbours of those concepts did not change over all runs. The aligned embeddings do not have any concepts with a stability of 1.0.

[BKM20] ran stability analysis for word embeddings across languages. They found that languages with higher morphological complexity tended to be less stable than languages with less complex morphology. The morphological complexity of a language depends on the relationships between the words in that language; more complex relationships means a higher morphological complexity [BBC15]. If we draw a parallel between the combined (AudioSet + Open Images) embeddings and language, it can be suggested that the combined embeddings have more complex relationships, so we would expect them to be less stable.

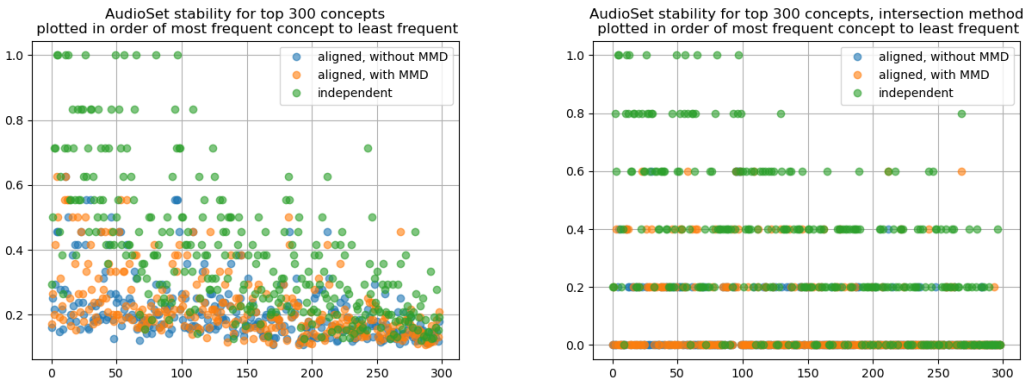

Figure 4.8: Stability of model variants for AudioSet. The y-axis is relative frequency of concepts (most frequent = 0). For embeddings in this domain produced by all model variants, there is a clear correlation between the relative frequency of a concept and the stability (however this effect is less pronounced for the aligned embeddings). Less frequent concepts are less stable.

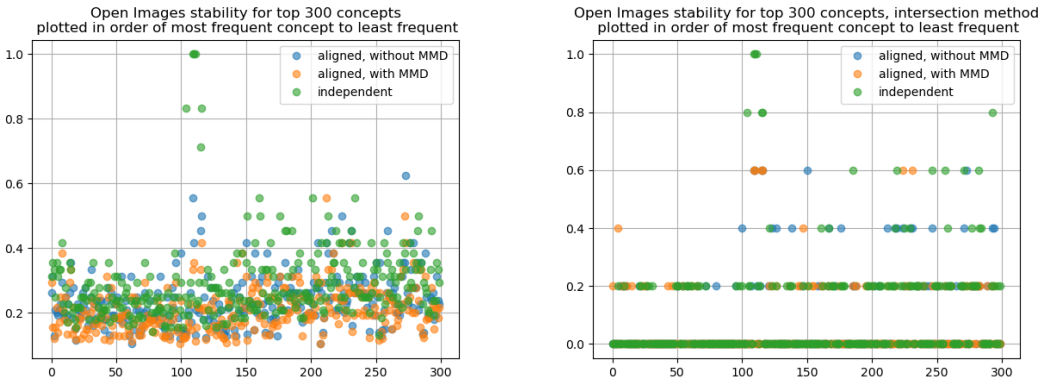

Figure 4.9: Stability of model variants for Open Images. The y-axis is relative frequency of concepts (most frequent = 0). There is not such a clear correlation between the relative frequency of a concept and the stability of its embedding, even for the independently learned embeddings.

4.4.5 Other findings

The role of MMD

Using MMD as a loss function has the effect of forcing $f(x)$ and y to have the same distribution, and $g(y)$ and x to have the same distribution. Therefore it is reasonable that it should have a positive effect on alignment accuracy. However if the weighting of the MMD loss is high relative to glove loss, there is more pressure for distribution matching but less pressure for good embeddings which can result in dysfunctional embeddings where all the concepts in the intersection are clustered together without regard for semantic similarity as shown in Figure 4.7.

Entropies and variances

As seen in Table 4.3, the entropies no longer correlate negatively with frequency of concepts, in the aligned embeddings. It is more appropriate to say they are now close to independent of the frequency of concepts, since the correlation is near zero.

GloVe loss scaling

If the GloVe loss was not scaled by the ratio of concepts (see Section 3.3.3), the effect of this was to cause the (intersecting) concepts in the embedding to exhibit poor semantic organization.

Chapter 5

Conclusions and further discussion

5.1 Summary of results

5.1.1 Restatement of project aims

The main aim was to learn jointly aligned probabilistic embeddings, represented by multidimensional Gaussian distributions with diagonal covariance, for concepts in the Open Images and AudioSet domains. Alignment and embedding learning were to happen concurrently. Independently learned probabilistic embeddings (training each domain separately) were also run as a baseline. The quality of these embeddings was examined by comparing the Spearman correlation of their pairwise similarities with three different human-curated similarity measures (only two for AudioSet due to lack of data)- MTURK-771 [Hal+12], WordNet [Mil95], and ILSVRC [RL20a]. Ideally, the aligned embeddings would be of higher quality (more similar with human judgement) than the independently learned embeddings.

5.1.2 Independently learned probabilistic embeddings

Clustering

The means of the probabilistic embeddings represent sensible clusters when viewed through scatter plots of t-SNE dimensionality reduction down to 2 dimensions. Qualitatively, items in the concept hierarchy that represent more abstract concepts from the point of view of the hierarchy (“Cat”, “Dog”) have nearest neighbours that are not very well clustered. Items that are more specific (“Domestic short-haired cat”) have nearest neighbours that

are more sensible. Examples of these phenomena are shown in Tables 3.1 and 3.2.

Statistics

The variances of the independently learned embeddings, when expressed as entropies, correlate negatively with co-occurrence frequency (Table 4.3). Concepts which occur less frequently in the input data have a higher variance. This is an expected result- there is more uncertainty about concepts that occur less frequently.

5.1.3 Semi-supervised learning of aligned embeddings

Alignment

Our algorithm successfully learned jointly aligned embeddings from two modalities, Open Images containing 19996 concepts and AudioSet containing 526 concepts, with an intersection of 230 concepts. Embeddings for both domains were learned simultaneously along with alignment, with no post-processing necessary to achieve alignment. The 230 intersecting concepts were used as semi-supervised input into the algorithm by tying their mapped values together during training as part of the loss function.

Alignment accuracy, as measured by the number of concepts in one domain whose nearest neighbour in the mapped domain was the true corresponding concept in the other domain (as described in Section 3.3.1), of more than 95% was obtained between the two domains, for the intersecting concepts. However, convergence to this level of accuracy while still maintaining sensible clustering (avoiding the degenerate case displayed in Figure 4.7) required some tuning of the model parameters, as described in Section 3.3.2

The criterion for saving the embeddings was when the mean alignment accuracy (of both OpenImages and AudioSet embeddings) was the highest. Knowing that the independently learned probabilistic embeddings for AudioSet required more epochs of training to converge than the similar case of Open Images, it is possible that the point of greatest alignment accuracy for AudioSet is at a different epoch than for Open Images, but that training to achieve this would cause a decrease in accuracy for Open Images.

Embedding quality

As a baseline, the aligned embeddings also displayed sensible semantic clustering (Figures 4.1 and 4.2). **Embedding quality as measured by Spearman correlation of embedding pair similarity and WordNet similarity measures was greater for the aligned Open Images embeddings than the independently learned Open Images embeddings.** When measuring by Spearman correlation of embedding pair similarity and ILSVRC / MTURK-771 datasets, embedding quality for the aligned Open Images was decreased compared to the independently learned embeddings.

For AudioSet embeddings, embedding quality as judged by the WordNet and ILSVRC metrics was greater for aligned than independently learned embeddings. A comparison with MTURK-771 was not run for AudioSet due to lack of data (only 3 pairs overlapped).

Including the empirical MMD statistic in the loss function increased alignment accuracy for both domains, but had varying effects on other measures of embedding quality. The MMD statistic increased embedding quality as measured by the Spearman correlation with WordNet similarity for the Open Images domain. For AudioSet and for Open Images compared with MTURK-771 and enhanced ILSVRC, MMD was either ineffective or decreased embedding quality.

The entropy of the aligned embeddings became decorrelated with the frequency of occurrence of each concept, compared to independently learned embeddings. The stability of aligned embeddings is also lower than that of independently learned embeddings, in that the top 5 nearest neighbours of concepts are more different over different runs.

We conclude that there is some evidence that aligned embeddings are of higher quality than independently learned embeddings, and therefore that learning from multiple modalities simultaneously helps generalise to better representations for both modalities. The results of comparing embedding quality based on WordNet lexical database similarity scores indicated that aligned embeddings are more closely correlated with human judgement. However, the comparisons with the two datasets using direct human-collected scores (MTURK-771 and enhanced ILSVRC) gave mixed results. Given that the latter two datasets are much smaller in comparison to WordNet, this bears further investigation.

5.2 Directions for future research

5.2.1 Controlling for different domain distributions

Alignment of embeddings from two different domains requires synchronising two distributions that have very different statistics. We already see that the sizes of the datasets are very unbalanced, and there is little overlap of concepts. To control for this, we could try splitting one dataset into 2 parts with a defined amount of overlap (for example, 2 subsets of 10000 concepts from Open Images, with an intersection of 2000 concepts) and trying to align those two subsets. This should allow us to learn about the alignment problem independently from the influence of the statistics of the different domains. We already encountered one issue which was that the GloVe loss needed to be scaled to achieve good convergence from the point of view of alignment accuracy.

5.2.2 Model parameters

As the problem and full loss function (Equation 3.3) are complex and contain many terms, they are likely to be very sensitive to initial conditions and choice of optimiser and learning rate. Cross-validation could be used to investigate better choices.

The use of alignment accuracy itself as a measure of convergence could be reconsidered for further experiments. We already see that with the use of MMD, alignment accuracy could increase but with a corresponding decrease in embedding quality; perhaps there is a way of including some measure of embedding quality in the stopping criteria or loss function.

5.2.3 Embedding dimensionality

Appropriate dimensionality of the embeddings for both domains should be investigated. As mentioned earlier, this number was chosen heuristically (keeping the ratio of concepts relative to the dimension the same as that of the original GloVe word embedding problem [PSM14]). Further studies should involve cross-validation to investigate appropriate dimensionality, which may not be the same for both domains. Tests with the independent embeddings showed that if dimensionality was too high (the example was dimension of 30 used for AudioSet embeddings), the resulting embeddings did not demonstrate good clustering as visualised through t-SNE.

5.2.4 Hierarchy of concepts

The highly non-overlapping concept sets mean that there are many concepts present in Open Images that are not specifically present in AudioSet. However, the Open Images concepts are actually at many levels of hierarchy. For example, there are different types of cat and different types of dog. Thus many of these may actually map to a single concept in AudioSet, and there is no provision for this at the moment.

Incorporating some form of hierarchy may lead to better results. This hierarchy may represented implicitly or explicitly. We saw that the nearest neighbours by distance to concepts that are higher up in the class hierarchy were less related than the nearest neighbours of concepts that are closer to the leaf nodes (Tables 3.1 and 3.2), and this is possibly because Euclidean distance does not represent hierarchy well [TH86]. Poincaré embeddings [NK17] use distance measures in non-Euclidean space (hyperbolic in this case) to allow hierarchical concepts to be represented implicitly.

Some explicit hierarchy information is available associated with the datasets, which we do not currently use. A partial such tree is shown in the figure below:

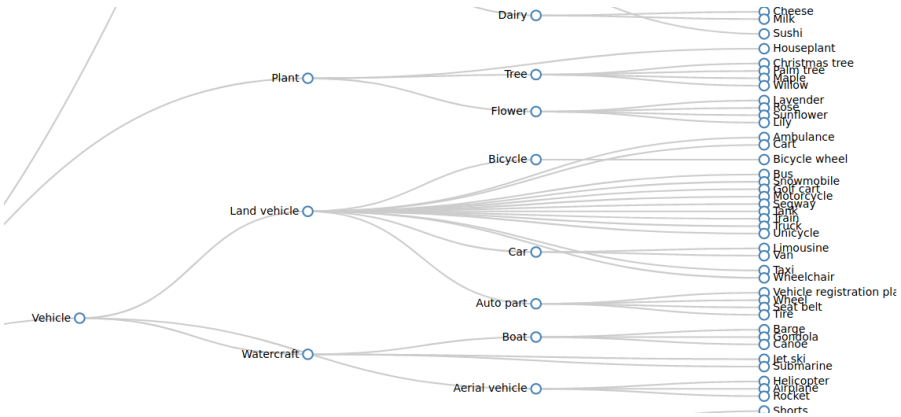

Figure 5.1: Tree structure showing part of the Open Images concept hierarchy. A similar structure exists for AudioSet.

From the t-SNE visuations of the independent embeddings as seen in Figures 3.2.2 and 3.2.2, it is clear that concept classes form clusters. We saw that the differences in arrangement of these clusters over different runs are not a simple spatial transformation (combining rotations, translations and stretching only). One obvious modification to the algorithm might be to align the parent concept classes for example “Cat” and “Dog” as a first pass, and then to align the subclasses around these anchor concepts. This process of

multiple alignment passes has been used in [GJB18] to perform alignment of cross-lingual word embeddings for bilingual lexicon induction.

There are many aural concepts that obviously will not have a visual representation, so this is yet another limitation of using these particular two modalities. Better results may perhaps be obtained by trying to align embeddings derived from Open Images and text, but then we would have to solve the following practical problems:

- How to resolve words to the Open Images namespace; one such possibility was used when running similarity comparisons with WordNet, described in the previous chapter. There is no sense information in the Open Images namespace, though it is noticeable that most of the categories are nouns.
- How to extract not just single words but phrases from the text, for example, "domestic short-haired cat". Computationally this can pose a problem as the vocabulary size of text corpora is already large, and now we would have to maintain n-grams as well to try and capture entire phrases.

5.2.5 Unsupervised learning of aligned embeddings

The experiments in this project lead to the ultimate goal of learning aligned embeddings in an unsupervised way, in keeping with the desire to emulate human learning. In this situation the concept universes for both domains are known, as are the items in the intersection. However the specific embeddings in the intersection in each domain would not be directly mapped in the losses during learning. Aggregate statistics of the domain intersection might be used, such as the MMD.

Preliminary work was unable to produce accuracies greater than 1% for unsupervised alignment, using the same model configuration as the semi-supervised case, only excluding the distance loss that related $||f(x) - y||$ and $||g(y) - x||$. It is possible that including some measure of graph-based similarity between the embeddings may increase convergence. The Friedman-Rafsky statistic described in [DK17] was tried, but there was no perceptible convergence.

Further preliminary work also tested the Manifold Alignment GAN [AK18] and the Wasserstein GAN [ACB17], using the configurations described in the respective papers, with the full loss function as in equation 3.3 used as the generator loss. Neither of these produced any feasible alignment. Mode collapse (all concepts mapping to the same embedding) was at first an issue, but even with using the minibatch discrimination technique

[Sal+16] to remove this possibility, no feasible alignment was found. The discriminator loss was asymptotically minimal, indicating that the discriminators were not able to tell the mapped values from the real values, but there was still no alignment. It is possible that there are simply no manifolds to align for the number of dimensions we have chosen (6) for our embeddings.

It is known that GANs do well on problems where the input data fall into specific discernible classes, and our dataset does not have this characteristic. Though there are discernible clusters, they are indistinct as befits a human taxonomy of concepts rather than visual representations of 10 digits, or works by different artists. Given this constraint, it is reasonable that a naive GAN might not work. In particular, there were different numbers of elements in the source and target domains; neither the MAGAN nor Wasserstein GAN were tested in this situation. Additionally, [Sal+16] found that the ILSVRC2012 dataset with 1000 categories was a challenge for their GAN because of the large number of classes, which is already fewer than the 19996 in our Open Images dataset. Much existing GAN research has been done on generating images matching certain criteria, for which input data is plentiful (some input sets number in the millions). In addition to having a very large number of classes, we also only have one example of each, so it should be considerably more difficult to learn this distribution with a GAN.

Lastly, previous experiments in the Love Lab [Ara20] found that the Wasserstein (Equation 2.1) and Sinkhorn (a regularised version of the Wasserstein) distances could be used as loss functions for unsupervised alignment of concept embeddings, but these experiments were run on synthetic data generated from Gaussian mixture models. Therefore it is conceivable that the Wasserstein and Sinkhorn distances could serve as loss functions for alignment of embeddings constructed from real data, with appropriate configuration.

Appendix A

A.1 Graph-based measures of statistical distance

We introduce the following notation, which differs from that in [DK17] to avoid overloading previously defined terms in this document.

- P is the distribution from which the points $X = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$ are drawn.
- Q is the distribution from which the points $Y = \{\boldsymbol{y}_1, ..., \boldsymbol{y}_n\}$ are drawn.
- $H(X) = (X, E)$ is the directed graph defined over the vertex set $X = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$, with edges E.
- $J(Y) = (Y, F)$ is the directed graph defined over the vertex set $Y = \{\boldsymbol{y}_1, ..., \boldsymbol{y}_m\}$, with edges F.
- These graphs are weighted with the distance function $d(\boldsymbol{x}, \boldsymbol{x}') = ||\boldsymbol{x} - \boldsymbol{x}'||$, and we will denote with $d(e)$ the weight of the edge e using distance function d.
- For a labelling of vertices $\pi : X \to \{1, 2\}$ and any edge e whose vertices i and j are adjacent, define $\Delta_\pi(e)$ to be 1 if e's end points have different labels under the mapping π.

The generic framework for the graph tests follows these steps:

1. Let Z be the union of the samples X and Y and let $K(Z)$ be the graph defined over all points. Define a mapping $\pi^* : Z \to \{1, 2\}$ such that $\pi^*(X) = 1$ and $\pi^*(Y) = 2$.
2. Use an algorithm A to choose a subset $U^* = A(K(Z))$ of the edges of Z, the idea being that this algorithm should encode some sort of neighbourhood structure.

- The Friedman-Rafsky test uses the minimum spanning tree of $H(X)$ as the algorithm for selecting the neighbourhood structure U^*.
- The k-nearest neighbours test adds an edge e to U^* if the starting point is one of the k nearest neighbours of the end point under the distance measure d.

3. The statistic $T_{\pi^*}(U^*) = \sum_{e \in U^*} \Delta_{\pi^*}(e)$ defines how many edges in U^* join points from X and Y.

4. If T_{π^*} is high, then many edges join points from X and Y, and X and Y are highly aligned. When using this statistic as a loss function, the negative of this must be minimised.

In order to use T_{π^*} in a loss function with backpropagation, we need to be able to calculate the derivatives $\frac{\partial T}{\partial \boldsymbol{x}_i}$ which normally do not exist. In [DK17] the strategy cited is to smooth these functions to make them continuously differentiable by turning them into expectations of probabilistic models in the exponential family.

We can express the optimal neighbourhood mapping U^* as

$$\begin{aligned} U^* = \underset{U \subseteq E}{\operatorname{argmin}} \sum_{e \in U} d(e) \quad \text{such that} \quad v(U) = 1 \\ \text{and further define} \quad \boldsymbol{d} \quad \text{to be the vector of edge weights} \quad d(e). \end{aligned} \tag{A.1}$$

where $v : 2^{|E|} \to \{0, 1\}$ is a mapping indicating if the set of edges is valid under the constraints of algorithm A, for example if each vertex has k neighbours for the KNN test, or if the set of edges forms a valid set of minimum spanning trees in the Friedman-Rafsky [FR79] test case.

The aim is to find a probability distribution over U whose expectation can be used in place of T_{π^*}. Without proof, we state the result from [DK17] that the following exponential family function suffices:

$$P(U|\boldsymbol{d}/\lambda) = \exp\Big[- \sum_{e \in U} d(e)/\lambda - A(-\boldsymbol{d}/\lambda \Big] v(U) \tag{A.2}$$

where λ is a hyperparameter (the "temperature parameter") and $A(-\boldsymbol{d}/\lambda)$ is the log-partition function that normalises the distribution. U^* is thus a maximum a posteriori configuration for $P(U|\boldsymbol{d}/\lambda)$, and as λ tends to 0, $P(U|\boldsymbol{d}/\lambda)$ will tend to the MAP estimate.

If we use the expectation $E_U[T_{\pi^*}(U)]$ in place of the original statistic $T_{\pi^*(U^*)}$, since $P(U)$ (A.2) is a member of the exponential family, we can compute its first and second moments, which lead to the values of the smoothed statistic as well as its derivative. These functions can now be used as loss functions of which minimisation corresponds to the two inputs having greater graph similarity.

Bibliography

[ACB17] Martin Arjovsky, Soumith Chintala, and Léon Bottou. *Wasserstein GAN*. 2017. arXiv: 1701.07875 [stat.ML].

[AK18] Matthew Amodio and Smita Krishnaswamy. *MAGAN: Aligning Biological Manifolds*. 2018. arXiv: 1803.00385 [cs.CV].

[Ara20] Kengo Arao. *Unsupervised Alignment of Concept Embeddings*. 2020.

[BBC15] Matthew Baerman, Dunstan Brown, and Greville Corbett. "Understanding and Measuring Morphological Complexity". In: *Understanding and Measuring Morphological Complexity* (Mar. 2015), pp. 1–240. DOI: 10.1093/acprof:oso/9780198723769.001.0001.

[BE21] Michael F. Bonner and Russell A. Epstein. "Object representations in the human brain reflect the co-occurrence statistics of vision and language". In: *Nature Communications* 12.1 (July 2021), p. 4081. ISSN: 2041-1723. DOI: 10.1038/s41467-021-24368-2. URL: https://doi.org/10.1038/s41467-021-24368-2.

[Bin+19] Eli Bingham et al. "Pyro: Deep Universal Probabilistic Programming". In: 20.1 (Jan. 2019), pp. 973–978. ISSN: 1532-4435.

[BKM20] Laura Burdick, Jonathan K. Kummerfeld, and Rada Mihalcea. *Analyzing the Surprising Variability in Word Embedding Stability Across Languages*. 2020. arXiv: 2004.14876 [cs.CL].

[BN03] Mikhail Belkin and Partha Niyogi. "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation". In: *Neural Comput.* 15.6 (June 2003), pp. 1373–1396. ISSN: 0899-7667. DOI: 10.1162/089976603321780317. URL: https://doi.org/10.1162/089976603321780317.

[Chu+21] Sanghyuk Chun et al. “Probabilistic Embeddings for Cross-Modal Retrieval”. In: *CoRR* abs/2101.05068 (2021). arXiv: 2101.05068. URL: https://arxiv.org/abs/2101.05068.

[Con+18] Alexis Conneau et al. *Word Translation Without Parallel Data*. 2018. arXiv: 1710.04087 [cs.CL].

[CV95] Corinna Cortes and Vladimir Vapnik. “Support-Vector Networks.” In: *Machine Learning* 20.3 (1995), pp. 273–297. URL: http://dblp.uni-trier.de/db/journals/ml/ml20.html#CortesV95.

[DK17] Josip Djolonga and Andreas Krause. *Learning Implicit Generative Models Using Differentiable Graph Tests*. 2017. arXiv: 1709.01006 [stat.ML].

[Fal19] et al. Falcon WA. “PyTorch Lightning”. In: *GitHub. Note: https://github.com/PyTorchLightnin lightning* 3 (2019).

[FR79] Jerome H. Friedman and Lawrence C. Rafsky. “Multivariate Generalizations of the Wald-Wolfowitz and Smirnov Two-Sample Tests”. In: *The Annals of Statistics* 7.4 (1979), pp. 697–717. DOI: 10.1214/aos/1176344722. URL: https://doi.org/10.1214/aos/1176344722.

[GBC16] Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. Cambridge, MA, USA: MIT Press, 2016.

[Gem+17] Jort F. Gemmeke et al. “Audio Set: An ontology and human-labeled dataset for audio events”. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017, pp. 776–780. DOI: 10.1109/ICASSP.2017.7952261.

[GH73] Ian E. Gordon and Sandra Hayward. “Second-order isomorphism of internal representations of familiar faces”. In: *Perception & Psychophysics* 14.2 (June 1973), pp. 334–336. ISSN: 1532-5962. DOI: 10.3758/BF03212400. URL: https://doi.org/10.3758/BF03212400.

[GJ90] Michael R. Garey and David S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. USA: W. H. Freeman & Co., 1990. ISBN: 0716710455.

[GJB18] Edouard Grave, Armand Joulin, and Quentin Berthet. “Unsupervised Alignment of Embeddings with Wasserstein Procrustes”. In: *CoRR* abs/1805.11222 (2018). arXiv: 1805.11222. URL: http://arxiv.org/abs/1805.11222.

[GJK17] Damien Garreau, Wittawat Jitkrittum, and Motonobu Kanagawa. "Large sample analysis of the median heuristic." In: *arXiv: Statistics Theory* (2017).

[Goo+14] Ian J. Goodfellow et al. *Generative Adversarial Networks*. 2014. arXiv: 1406.2661 [stat.ML].

[Goo91] C. Goodall. "Procrustes methods in the statistical analysis of shape". In: *Journal of the royal statistical society series b-methodological* 53 (1991), pp. 285–321.

[GR02] Robert L. Goldstone and Brian J. Rogosky. "Using relations within conceptual systems to translate across conceptual systems". In: *Cognition* 84.3 (2002), pp. 295–320. ISSN: 0010-0277. DOI: https://doi.org/10.1016/S0010-0277(02)00053-7. URL: https://www.sciencedirect.com/science/article/pii/S0010027702000537.

[Gre+12] Arthur Gretton et al. "A Kernel Two-Sample Test". In: *J. Mach. Learn. Res.* 13.null (Mar. 2012), pp. 723–773. ISSN: 1532-4435.

[Hal+12] Guy Halawi et al. "Large-scale learning of word relatedness with constraints". In: (Aug. 2012). DOI: 10.1145/2339530.2339751.

[Hei+20] Nicolas Heist et al. "Knowledge Graphs on the Web - an Overview". In: *CoRR* abs/2003.00719 (2020). arXiv: 2003.00719. URL: https://arxiv.org/abs/2003.00719.

[Jou+18] Armand Joulin et al. "Improving Supervised Bilingual Mapping of Word Embeddings". In: *CoRR* abs/1804.07745 (2018). arXiv: 1804.07745. URL: http://arxiv.org/abs/1804.07745.

[KA20] Alexander Kalinowski and Yuan An. *A Survey of Embedding Space Alignment Methods for Language and Knowledge Graphs*. 2020. arXiv: 2010.13688 [cs.CL].

[KB17] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG].

[Kuz+18] Alina Kuznetsova et al. "The Open Images Dataset V4: Unified image classification, object detection, and visual relationship detection at scale". In: *CoRR* abs/1811.00982 (2018). arXiv: 1811.00982. URL: http://arxiv.org/abs/1811.00982.

[LDB15] A. Lazaridou, Georgiana Dinu, and Marco Baroni. “Hubness and Pollution: Delving into Cross-Space Mapping for Zero-Shot Learning”. In: *ACL*. 2015.

[Len95] Douglas B. Lenat. “CYC: A Large-Scale Investment in Knowledge Infrastructure”. In: *Commun. ACM* 38.11 (Nov. 1995), pp. 33–38. ISSN: 0001-0782. DOI: 10.1145/219717.219745. URL: https://doi.org/10.1145/219717.219745.

[LMC98] Claudia Leacock, George A. Miller, and Martin Chodorow. “Using Corpus Statistics and WordNet Relations for Sense Identification”. In: *Comput. Linguist.* 24.1 (Mar. 1998), pp. 147–165. ISSN: 0891-2017.

[MH08] Laurens van der Maaten and Geoffrey E. Hinton. “Visualizing High-Dimensional Data Using t-SNE”. In: *Journal of Machine Learning Research* 9 (2008), pp. 2579–2605.

[Mik+13] Tomas Mikolov et al. *Efficient Estimation of Word Representations in Vector Space*. 2013. arXiv: 1301.3781 [cs.CL].

[Mil95] George A. Miller. “WordNet: A Lexical Database for English”. In: *Commun. ACM* 38.11 (Nov. 1995), pp. 39–41. ISSN: 0001-0782. DOI: 10.1145/219717.219748. URL: https://doi.org/10.1145/219717.219748.

[ML21] Eric Margolis and Stephen Laurence. “Concepts”. In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Spring 2021. Metaphysics Research Lab, Stanford University, 2021.

[MLS13] Tomás Mikolov, Quoc V. Le, and Ilya Sutskever. “Exploiting Similarities among Languages for Machine Translation”. In: *CoRR* abs/1309.4168 (2013). arXiv: 1309.4168. URL: http://arxiv.org/abs/1309.4168.

[Mua+17] Krikamol Muandet et al. “Kernel Mean Embedding of Distributions: A Review and Beyond”. In: *Foundations and Trends® in Machine Learning* 10.1-2 (2017), pp. 1–141. ISSN: 1935-8245. DOI: 10.1561/2200000060. URL: http://dx.doi.org/10.1561/2200000060.

[NK17] Maximilian Nickel and Douwe Kiela. “Poincaré Embeddings for Learning Hierarchical Representations”. In: *CoRR* abs/1705.08039 (2017). arXiv: 1705.08039. URL: http://arxiv.org/abs/1705.08039.

[Pas+19] Adam Paszke et al. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[Pin07] Steven Pinker. *The Language Instinct (1994/2007)*. New York, NY: Harper Perennial Modern Classics., 2007.

[PP96] S. Pinker and Alan S. Prince. “The Nature of Human Concepts/Evidence from an Unusual Source”. In: *Communication and Cognition. Monographies* 29 (1996), pp. 307–361.

[PSM14] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. “GloVe: Global Vectors for Word Representation”. In: *Empirical Methods in Natural Language Processing (EMNLP)*. 2014, pp. 1532–1543. URL: http://www.aclweb.org/anthology/D14-1162.

[RHW86] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. “Learning Internal Representations by Error Propagation”. In: *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*. Cambridge, MA, USA: MIT Press, 1986, pp. 318–362. ISBN: 026268053X.

[RL20a] Brett D. Roads and Bradley C. Love. “Enriching ImageNet with Human Similarity Judgments and Psychological Embeddings”. In: *CoRR* abs/2011.11015 (2020). arXiv: 2011.11015. URL: https://arxiv.org/abs/2011.11015.

[RL20b] Brett D. Roads and Bradley C. Love. “Learning as the unsupervised alignment of conceptual systems”. In: *Nature Machine Intelligence* 2.1 (Jan. 2020), pp. 76–82. ISSN: 2522-5839. DOI: 10.1038/s42256-019-0132-2. URL: https://doi.org/10.1038/s42256-019-0132-2.

[RS00] Sam T. Roweis and Lawrence K. Saul. “Nonlinear Dimensionality Reduction by Locally Linear Embedding”. In: *Science* 290.5500 (2000), pp. 2323–2326. ISSN: 0036-8075. DOI: 10.1126/science.290.5500.2323. eprint: https://science.sciencemag.org/content/290/5500/2323.full.pdf. URL: https://science.sciencemag.org/content/290/5500/2323.

[Rud17] Sebastian Ruder. “An overview of multi-task learning in deep neural networks”. In: (2017). arXiv: 1706.05098. URL: https://arxiv.org/abs/1706.05098.

[Sal+16] Tim Salimans et al. “Improved Techniques for Training GANs”. In: *CoRR* abs/1606.03498 (2016). arXiv: 1606.03498. URL: http://arxiv.org/abs/1606.03498.

[SC70] Roger N Shepard and Susan Chipman. “Second-order isomorphism of internal representations: Shapes of states”. In: *Cognitive Psychology* 1.1 (1970), pp. 1–17. ISSN: 0010-0285. DOI: https://doi.org/10.1016/0010-0285(70)90002-2. URL: https://www.sciencedirect.com/science/article/pii/0010028570900022.

[SLS15] Jaeyong Sung, Ian Lenz, and Ashutosh Saxena. “Deep Multimodal Embedding: Manipulating Novel Objects with Point-clouds, Language and Trajectories”. In: *CoRR* abs/1509.07831 (2015). arXiv: 1509.07831. URL: http://arxiv.org/abs/1509.07831.

[SNG13] Dustin E. Stansbury, Thomas Naselaris, and Jack L. Gallant. “Natural Scene Statistics Account for the Representation of Scene Categories in Human Visual Cortex”. In: *Neuron* 79.5 (2013), pp. 1025–1034. ISSN: 0896-6273. DOI: https://doi.org/10.1016/j.neuron.2013.06.034. URL: https://www.sciencedirect.com/science/article/pii/S0896627313005503.

[TH86] Amos Tversky and J Hutchinson. “Nearest neighbor analysis of psychological spaces.” In: *Psychological Review* 93.1 (1986), pp. 3–22.

[TSL00] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. “A Global Geometric Framework for Nonlinear Dimensionality Reduction”. In: *Science* 290.5500 (2000), pp. 2319–2323. ISSN: 0036-8075. DOI: 10.1126/science.290.5500.2319. eprint: https://science.sciencemag.org/content/290/5500/2319.full.pdf. URL: https://science.sciencemag.org/content/290/5500/2319.

[VM15] Luke Vilnis and Andrew McCallum. *Word Representations via Gaussian Embedding*. 2015. arXiv: 1412.6623 [cs.CL].

[WPM11] Chang Wang, Krafftm Peter, and Sridhar Mahadevan. *Manifold Learning Theory and Applications*. 1st. USA: CRC Press, Inc., 2011. ISBN: 1439871094.

[YM05] Robert L. Goldstone Ying Feng and Vladimir Menkov. “A graph matching algorithm and its application to conceptual system translation”. In: *International Journal on Artificial Intelligence Tools* 14.01n02 (2005), pp. 77–99. DOI:

https://doi.org/10.1142/S0218213005002004. URL: https://www.worldscientific.com/doi/abs/10.1142/S0218213005002004.

[Zha+17] Meng Zhang et al. "Earth Mover's Distance Minimization for Unsupervised Bilingual Lexicon Induction". In: *EMNLP*. 2017.

[Zho+16] Bolei Zhou et al. "Semantic Understanding of Scenes through the ADE20K Dataset". In: *CoRR* abs/1608.05442 (2016). arXiv: 1608.05442. URL: http://arxiv.org/abs/1608.05442.

[Zho+19] Chunting Zhou et al. "Density Matching for Bilingual Word Embedding". In: *CoRR* abs/1904.02343 (2019). arXiv: 1904.02343. URL: http://arxiv.org/abs/1904.02343.

[Zhu+17] Jun-Yan Zhu et al. "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks". In: *CoRR* abs/1703.10593 (2017). arXiv: 1703.10593. URL: http://arxiv.org/abs/1703.10593.

[Zhu+19] Hao Zhu et al. "A Review of Point Set Registration: From Pairwise Registration to Groupwise Registration". In: *Sensors (Basel, Switzerland)* 19 (2019).