Multi-task Learning of Pairwise Sequence Classification Tasks Over Disparate Label Spaces

Isabelle Augenstein[1]*, **Sebastian Ruder**[23]*, **Anders Søgaard**[1]
[1]Department of Computer Science, University of Copenhagen, Denmark
[2]Insight Research Centre, National University of Ireland, Galway
[3]Aylien Ltd., Dublin, Ireland
{augenstein|soegaard}@di.ku.dk, sebastian@ruder.io

Abstract

We combine multi-task learning and semi-supervised learning by inducing a joint embedding space between disparate label spaces and learning transfer functions between label embeddings, enabling us to jointly leverage unlabelled data and auxiliary, annotated datasets. We evaluate our approach on a variety of sequence classification tasks with disparate label spaces. We outperform strong single and multi-task baselines and achieve a new state-of-the-art for topic-based sentiment analysis.

1 Introduction

Multi-task learning (MTL) and semi-supervised learning are both successful paradigms for learning in scenarios with limited labelled data and have in recent years been applied to almost all areas of NLP. Applications of MTL in NLP, for example, include partial parsing (Søgaard and Goldberg, 2016), text normalisation (Bollman et al., 2017), neural machine translation (Luong et al., 2016), and keyphrase boundary classification (Augenstein and Søgaard, 2017).

Contemporary work in MTL for NLP typically focuses on learning representations that are useful across tasks, often through hard parameter sharing of hidden layers of neural networks (Collobert et al., 2011; Søgaard and Goldberg, 2016). If tasks share optimal hypothesis classes at the level of these representations, MTL leads to improvements (Baxter, 2000). However, while sharing hidden layers of neural networks is an effective regulariser (Søgaard and Goldberg, 2016), we potentially *loose synergies between the classification functions* trained to associate these representations with class labels. This paper sets out to build an architecture in which such synergies are exploited, with an application to pairwise sequence classification tasks. Doing so, we achieve a new state of the art on topic-based sentiment analysis.

For many NLP tasks, disparate label sets are weakly correlated, e.g. part-of-speech tags correlate with dependencies (Hashimoto et al., 2017), sentiment correlates with emotion (Felbo et al., 2017; Eisner et al., 2016), etc. We thus propose to induce a joint label embedding space (visualised in Figure 2) using a Label Embedding Layer that allows us to model these relationships, which we show helps with learning.

In addition, for tasks where labels are closely related, we should be able to not only model their relationship, but also to directly estimate the corresponding label of the target task based on auxiliary predictions. To this end, we propose to train a Label Transfer Network (LTN) jointly with the model to produce pseudo-labels across tasks.

The LTN can be used to label unlabelled and auxiliary task data by utilising the 'dark knowledge' (Hinton et al., 2015) contained in auxiliary model predictions. This pseudo-labelled data is then incorporated into the model via semi-supervised learning, leading to a natural combination of multi-task learning and semi-supervised learning. We additionally augment the LTN with data-specific diversity features (Ruder and Plank, 2017) that aid in learning.

Contributions Our contributions are: a) We model the relationships between labels by inducing a joint label space for multi-task learning. b) We propose a Label Transfer Network that learns to transfer labels between tasks and propose to use semi-supervised learning to leverage them for training. c) We evaluate MTL approaches on a variety of classification tasks and shed new light on settings where multi-task learning works. d) We perform an extensive ablation study of our model.

*The first two authors contributed equally.

Proceedings of NAACL-HLT 2018, pages 1896–1906
New Orleans, Louisiana, June 1 - 6, 2018.

e) We report state-of-the-art performance on topic-based sentiment analysis.

2 Related work

Learning task similarities Existing approaches for learning similarities between tasks enforce a clustering of tasks (Evgeniou et al., 2005; Jacob et al., 2009), induce a shared prior (Yu et al., 2005; Xue et al., 2007; Daumé III, 2009), or learn a grouping (Kang et al., 2011; Kumar and Daumé III, 2012). These approaches focus on homogeneous tasks and employ linear or Bayesian models. They can thus not be directly applied to our setting with tasks using disparate label sets.

Multi-task learning with neural networks Recent work in multi-task learning goes beyond hard parameter sharing (Caruana, 1993) and considers different sharing structures, e.g. only sharing at lower layers (Søgaard and Goldberg, 2016) and induces private and shared subspaces (Liu et al., 2017; Ruder et al., 2017). These approaches, however, are not able to take into account relationships between labels that may aid in learning. Another related direction is to train on disparate annotations of the same task (Chen et al., 2016; Peng et al., 2017). In contrast, the different nature of our tasks requires a modelling of their label spaces.

Semi-supervised learning There exists a wide range of semi-supervised learning algorithms, e.g., self-training, co-training, tri-training, EM, and combinations thereof, several of which have also been used in NLP. Our approach is probably most closely related to an algorithm called *co-forest* (Li and Zhou, 2007). In co-forest, like here, each learner is improved with unlabeled instances labeled by the ensemble consisting of all the other learners. Note also that several researchers have proposed using auxiliary tasks that are unsupervised (Plank et al., 2016; Rei, 2017), which also leads to a form of semi-supervised models.

Label transformations The idea of manually mapping between label sets or learning such a mapping to facilitate transfer is not new. Zhang et al. (2012) use distributional information to map from a language-specific tagset to a tagset used for other languages, in order to facilitate cross-lingual transfer. More related to this work, Kim et al. (2015) use canonical correlation analysis to transfer between tasks with disparate label spaces. There has also been work on label transformations in the context of multi-label classification problems (Yeh et al., 2017).

3 Multi-task learning with disparate label spaces

3.1 Problem definition

In our multi-task learning scenario, we have access to labelled datasets for T tasks $\mathcal{T}_1, \ldots, \mathcal{T}_T$ at training time with a target task $\mathcal{T}_T$ that we particularly care about. The training dataset for task $\mathcal{T}_i$ consists of N_k examples $X_{\mathcal{T}_i} = \{x_1^{\mathcal{T}_i}, \ldots, x_{N_k}^{\mathcal{T}_i}\}$ and their labels $Y_{\mathcal{T}_i} = \{\mathbf{y}_1^{\mathcal{T}_i}, \ldots, \mathbf{y}_{N_k}^{\mathcal{T}_i}\}$. Our base model is a deep neural network that performs classic hard parameter sharing (Caruana, 1993): It shares its parameters across tasks and has task-specific softmax output layers, which output a probability distribution $\mathbf{p}^{\mathcal{T}_i}$ for task $\mathcal{T}_i$ according to the following equation:

$$\mathbf{p}^{\mathcal{T}_i} = \mathrm{softmax}(\mathbf{W}^{\mathcal{T}_i}\mathbf{h} + \mathbf{b}^{\mathcal{T}_i}) \quad (1)$$

where $\mathrm{softmax}(\mathbf{x}) = e^{\mathbf{x}} / \sum_{i=1}^{\|\mathbf{x}\|} e^{\mathbf{x}_i}$, $\mathbf{W}^{\mathcal{T}_i} \in \mathbb{R}^{L_i \times h}$, $\mathbf{b}^{\mathcal{T}_i} \in \mathbb{R}^{L_i}$ is the weight matrix and bias term of the output layer of task $\mathcal{T}_i$ respectively, $\mathbf{h} \in \mathbb{R}^h$ is the jointly learned hidden representation, L_i is the number of labels for task $\mathcal{T}_i$, and h is the dimensionality of $\mathbf{h}$.

The MTL model is then trained to minimise the sum of the individual task losses:

$$\mathcal{L} = \lambda_1 \mathcal{L}_1 + \ldots + \lambda_T \mathcal{L}_T \quad (2)$$

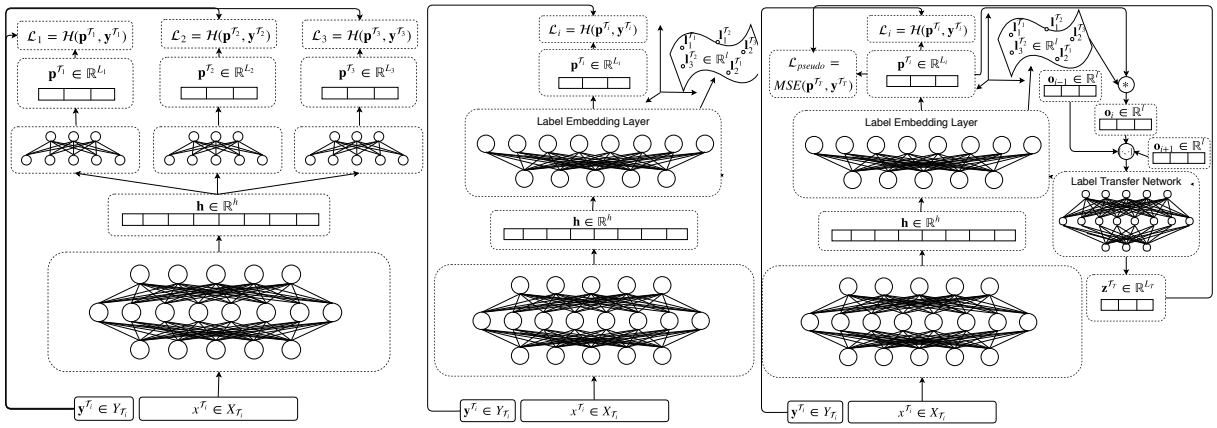
where $\mathcal{L}_i$ is the negative log-likelihood objective $\mathcal{L}_i = \mathcal{H}(\mathbf{p}^{\mathcal{T}_i}, \mathbf{y}^{\mathcal{T}_i}) = -\frac{1}{N} \sum_n \sum_j \log \mathbf{p}_j^{\mathcal{T}_i} \mathbf{y}_j^{\mathcal{T}_i}$ and λ_i is a parameter that determines the weight of task $\mathcal{T}_i$. In practice, we apply the same weight to all tasks. We show the full set-up in Figure 1a.

3.2 Label Embedding Layer

In order to learn the relationships between labels, we propose a Label Embedding Layer (LEL) that embeds the labels of all tasks in a joint space. Instead of training separate softmax output layers as above, we introduce a label compatibility function $c(\cdot, \cdot)$ that measures how similar a label with embedding $\mathbf{l}$ is to the hidden representation $\mathbf{h}$:

$$c(\mathbf{l}, \mathbf{h}) = \mathbf{l} \cdot \mathbf{h} \quad (3)$$

where $\cdot$ is the dot product. This is similar to the Universal Schema Latent Feature Model introduced by Riedel et al. (2013). In contrast to



(a) Multi-task learning (b) MTL with LEL (c) Semi-supervised MTL with LTN

Figure 1: a) Multi-task learning (MTL) with hard parameter sharing and 3 tasks $\mathcal{T}_{1-3}$ and L_{1-3} labels per task. A shared representation $\mathbf{h}$ is used as input to task-specific softmax layers, which optimise cross-entropy losses $\mathcal{L}_{1-3}$. b) MTL with the Label Embedding Layer (LEL) embeds task labels $\mathbf{l}_{1-L_i}^{\mathcal{T}_{1-3}}$ in a joint embedding space and uses these for prediction with a label compatibility function. c) Semi-supervised MTL with the Label Transfer Network (LTN) in addition optimises an unsupervised loss $\mathcal{L}_{pseudo}$ over pseudo-labels $\mathbf{z}^{\mathcal{T}_T}$ on auxiliary/unlabelled data. The pseudo-labels $\mathbf{z}^{\mathcal{T}_T}$ are produced by the LTN for the main task $\mathcal{T}_T$ using the concatenation of auxiliary task label output embeddings $[\mathbf{o}_{i-1}, \mathbf{o}_i, \mathbf{o}_{i+1}]$ as input.

other models that use the dot product in the objective function, we do not have to rely on negative sampling and a hinge loss (Collobert and Weston, 2008) as negative instances (labels) are known. For efficiency purposes, we use matrix multiplication instead of a single dot product and softmax instead of sigmoid activations:

$$\mathbf{p} = \text{softmax}(\mathbf{L}\mathbf{h}) \tag{4}$$

where $\mathbf{L} \in \mathbb{R}^{(\sum_i L_i) \times l}$ is the label embedding matrix for all tasks and l is the dimensionality of the label embeddings. In practice, we set l to the hidden dimensionality h. We use padding if $l < h$. We apply a task-specific mask to $\mathbf{L}$ in order to obtain a task-specific probability distribution $\mathbf{p}^{\mathcal{T}_i}$. The LEL is shared across all tasks, which allows us to learn the relationships between the labels in the joint embedding space. We show MTL with the LEL in Figure 1b.

3.3 Label Transfer Network

The LEL allows us to learn the relationships between labels. In order to make use of these relationships, we would like to leverage the predictions of our auxiliary tasks to estimate a label for the target task. To this end, we introduce the Label Transfer Network (LTN). This network takes the auxiliary task outputs as input. In particular, we define the output label embedding $\mathbf{o}_i$ of task $\mathcal{T}_i$ as the sum of the task's label embeddings $\mathbf{l}_j$ weighted with their probability $\mathbf{p}_j^{\mathcal{T}_i}$:

$$\mathbf{o}_i = \sum_{j=1}^{L_i} \mathbf{p}_j^{\mathcal{T}_i} \mathbf{l}_j \tag{5}$$

The label embeddings $\mathbf{l}$ encode general relationship between labels, while the model's probability distribution $\mathbf{p}^{\mathcal{T}_i}$ over its predictions encodes fine-grained information useful for learning (Hinton et al., 2015). The LTN is trained on labelled target task data. For each example, the corresponding label output embeddings of the auxiliary tasks are fed into a multi-layer perceptron (MLP), which is trained with a negative log-likelihood objective $\mathcal{L}_{\text{LTN}}$ to produce a pseudo-label $\mathbf{z}^{\mathcal{T}_T}$ for the target task $\mathcal{T}_T$:

$$\text{LTN}_T = \text{MLP}([\mathbf{o}_1, \ldots, \mathbf{o}_{T-1}]) \tag{6}$$

where $[\cdot, \cdot]$ designates concatenation. The mapping of the tasks in the LTN yields another signal that can be useful for optimisation and act as a regulariser. The LTN can also be seen as a mixture-of-experts layer (Jacobs et al., 1991) where the experts are the auxiliary task models. As the label embeddings are learned jointly with the main model, the LTN is more sensitive to the relationships between labels than a separately learned mixture-of-experts model that only relies on the experts' output distributions. As such, the LTN

can be directly used to produce predictions on unseen data.

3.4 Semi-supervised MTL

The downside of the LTN is that it requires additional parameters and relies on the predictions of the auxiliary models, which impacts the runtime during testing. Instead, of using the LTN for prediction directly, we can use it to provide pseudo-labels for unlabelled or auxiliary task data by utilising auxiliary predictions for semi-supervised learning.

We train the target task model on the pseudo-labelled data to minimise the squared error between the model predictions $\mathbf{p}^{\mathcal{T}_i}$ and the pseudo labels $\mathbf{z}^{\mathcal{T}_i}$ produced by the LTN:

$$\mathcal{L}_{pseudo} = MSE(\mathbf{p}^{\mathcal{T}_T}, \mathbf{z}^{\mathcal{T}_T}) = ||\mathbf{p}^{\mathcal{T}_T} - \mathbf{z}^{\mathcal{T}_T}||^2 \tag{7}$$

We add this loss term to the MTL loss in Equation 2. As the LTN is learned together with the MTL model, pseudo-labels produced early during training will likely not be helpful as they are based on unreliable auxiliary predictions. For this reason, we first train the base MTL model until convergence and then augment it with the LTN. We show the full semi-supervised learning procedure in Figure 1c.

3.5 Data-specific features

When there is a domain shift between the datasets of different tasks as is common for instance when learning NER models with different label sets, the output label embeddings might not contain sufficient information to bridge the domain gap.

To mitigate this discrepancy, we augment the LTN's input with features that have been found useful for transfer learning (Ruder and Plank, 2017). In particular, we use the number of word types, type-token ratio, entropy, Simpson's index, and Rényi entropy as diversity features. We calculate each feature for each example.[1] The features are then concatenated with the input of the LTN.

3.6 Other multi-task improvements

Hard parameter sharing can be overly restrictive and provide a regularisation that is too heavy when jointly learning many tasks. For this reason, we propose several additional improvements that seek to alleviate this burden: We use skip-connections, which have been shown to be useful for multi-task learning in recent work (Ruder et al., 2017). Furthermore, we add a task-specific layer before the output layer, which is useful for learning task-specific transformations of the shared representations (Søgaard and Goldberg, 2016; Ruder et al., 2017).

[1]For more information regarding the feature calculation, refer to Ruder and Plank (2017).

Task	Domain	N	L	Metric
`Topic-2`	Twitter	4,346	2	ρ^{PN}
`Topic-5`	Twitter	6,000	5	MAE^{M}
`Target`	Twitter	6,248	3	F_1^{M}
`Stance`	Twitter	2,914	3	F_1^{FA}
`ABSA-L`	Reviews	2,909	3	Acc
`ABSA-R`	Reviews	2,507	3	Acc
`FNC-1`	News	39,741	4	Acc
`MultiNLI`	Diverse	392,702	3	Acc

Table 1: Training set statistics and evaluation metrics of every task. N: # of examples. L: # of labels.

4 Experiments

For our experiments, we evaluate on a wide range of text classification tasks. In particular, we choose pairwise classification tasks—i.e. those that condition the reading of one sequence on another sequence—as we are interested in understanding if knowledge can be transferred even for these more complex interactions. To the best of our knowledge, this is the first work on transfer learning between such pairwise sequence classification tasks. We implement all our models in Tensorflow (Abadi et al., 2016) and release the code at `https://github.com/coastalcph/mtl-disparate`.

4.1 Tasks and datasets

We use the following tasks and datasets for our experiments, show task statistics in Table 1, and summarise examples in Table 2:

Topic-based sentiment analysis Topic-based sentiment analysis aims to estimate the sentiment of a tweet known to be about a given topic. We use the data from SemEval-2016 Task 4 Subtask B and C (Nakov et al., 2016) for predicting on a two-point scale of positive and negative (`Topic-2`) and five-point scale ranging from highly negative to highly positive (`Topic-5`) respectively. An example from this dataset would be to classify the

Topic-based sentiment analysis: *Tweet*: No power at home, sat in the dark listening to AC/DC in the hope it'll make the electricity come back again *Topic*: AC/DC *Label*: positive
Target-dependent sentiment analysis: *Text*: how do you like settlers of catan for the wii? *Target*: wii *Label*: neutral
Aspect-based sentiment analysis: *Text*: For the price, you cannot eat this well in Manhattan *Aspects*: restaurant prices, food quality *Label*: positive
Stance detection: *Tweet*: Be prepared - if we continue the policies of the liberal left, we will be #Greece *Target*: Donald Trump *Label*: favor
Fake news detection: *Document*: Dino Ferrari hooked the whopper wels catfish, (...), which could be the biggest in the world. *Headline*: Fisherman lands 19 STONE catfish which could be the biggest in the world to be hooked *Label*: agree
Natural language inference: *Premise*: Fun for only children *Hypothesis*: Fun for adults and children *Label*: contradiction

Table 2: Example instances from the datasets described in Section 4.1.

tweet "No power at home, sat in the dark listening to AC/DC in the hope it'll make the electricity come back again" known to be about the topic "AC/DC", which is labelled as a positive sentiment. The evaluation metrics for `Topic-2` and `Topic-5` are macro-averaged recall (ρ^{PN}) and macro-averaged mean absolute error (MAE^M) respectively, which are both averaged across topics.

Target-dependent sentiment analysis Target-dependent sentiment analysis (`Target`) seeks to classify the sentiment of a text's author towards an entity that occurs in the text as positive, negative, or neutral. We use the data from Dong et al. (2014). An example instance is the expression "how do you like settlers of catan for the wii?" which is labelled as neutral towards the target "wii'.' The evaluation metric is macro-averaged F_1 (F_1^M).

Aspect-based sentiment analysis Aspect-based sentiment analysis is the task of identifying whether an aspect, i.e. a particular property of an item is associated with a positive, negative, or neutral sentiment (Ruder et al., 2016). We use the data of SemEval-2016 Task 5 Subtask 1 Slot 3 (Pontiki et al., 2016) for the laptops (`ABSA-L`) and restaurants (`ABSA-R`) domains. An example is the sentence "For the price, you cannot eat this well in Manhattan", labelled as positive towards both the aspects "restaurant prices" and "food quality". The evaluation metric for both domains is accuracy (Acc).

Stance detection Stance detection (`Stance`) requires a model, given a text and a target entity, which might not appear in the text, to predict whether the author of the text is in favour or against the target or whether neither inference is likely (Augenstein et al., 2016). We use the data of SemEval-2016 Task 6 Subtask B (Mohammad et al., 2016). An example from this dataset would be to predict the stance of the tweet "Be prepared - if we continue the policies of the liberal left, we will be #Greece" towards the topic "Donald Trump", labelled as "favor". The evaluation metric is the macro-averaged F_1 score of the "favour" and "against" classes (F_1^{FA}).

Fake news detection The goal of fake news detection in the context of the Fake News Challenge[2] is to estimate whether the body of a news article agrees, disagrees, discusses, or is unrelated towards a headline. We use the data from the first stage of the Fake News Challenge (`FNC-1`). An example for this dataset is the document "Dino Ferrari hooked the whopper wels catfish, (...), which could be the biggest in the world." with the headline "Fisherman lands 19 STONE catfish which could be the biggest in the world to be hooked" labelled as "agree". The evaluation metric is accuracy (Acc)[3].

Natural language inference Natural language inference is the task of predicting whether one sentences entails, contradicts, or is neutral towards another one. We use the Multi-Genre NLI corpus (`MultiNLI`) from the RepEval 2017 shared task (Nangia et al., 2017). An example for an instance would be the sentence pair "Fun for only children", "Fun for adults and children", which are in a "contradiction" relationship. The evaluation metric is accuracy (Acc).

[2] `http://www.fakenewschallenge.org/`

[3] We use the same metric as Riedel et al. (2017).

	Stance	FNC	MultiNLI	Topic-2	Topic-5*	ABSA-L	ABSA-R	Target
Augenstein et al. (2016)	**49.01**	-	-	-	-	-	-	-
Riedel et al. (2017)	-	**88.46**	-	-	-	-	-	-
Chen et al. (2017)	-	-	**74.90**	-	-	-	-	-
Palogiannidi et al. (2016)	-	-	-	79.90	-	-	-	-
Balikas and Amini (2016)	-	-	-	-	**0.719**	-	-	-
Brun et al. (2016)	-	-	-	-	-	-	**88.13**	-
Kumar et al. (2016)	-	-	-	-	-	**82.77**	86.73	-
Vo and Zhang (2015)	-	-	-	-	-	-	-	**69.90**
STL	41.1	72.72	49.25	63.92	0.919	76.74	67.47	64.01
MTL + LEL	46.26	72.71	49.94	**80.52**	0.814	74.94	79.90	66.42
MTL + LEL + LTN, main model	43.16	72.73	48.75	73.90	0.810	75.06	83.71	66.10
MTL + LEL + LTN + semi, main model	43.56	72.72	48.00	72.35	0.821	75.42	83.26	63.00

Table 3: Comparison of our best performing models on the test set against a single task baseline and the state of the art, with task specific metrics. *: lower is better. Bold: best. Underlined: second-best.

4.2 Base model

Our base model is the Bidirectional Encoding model (Augenstein et al., 2016), a state-of-the-art model for stance detection that conditions a bidirectional LSTM (BiLSTM) encoding of a text on the BiLSTM encoding of the target. Unlike Augenstein et al. (2016), we do not pre-train word embeddings on a larger set of unlabelled in-domain text for each task as we are mainly interested in exploring the benefit of multi-task learning for generalisation.

4.3 Training settings

We use BiLSTMs with one hidden layer of 100 dimensions, 100-dimensional randomly initialised word embeddings, a label embedding size of 100. We train our models with RMSProp, a learning rate of 0.001, a batch size of 128, and early stopping on the validation set of the main task with a patience of 3.

5 Results

Our main results are shown in Table 3, with a comparison against the state of the art. We present the results of our multi-task learning network with label embeddings (MTL + LEL), multi-task learning with label transfer (MTL + LEL + LTN), and the semi-supervised extension of this model. On 7/8 tasks, at least one of our architectures is better than single-task learning; and in 4/8, all our architectures are much better than single-task learning.

The state-of-the-art systems we compare against are often highly specialised, task-dependent architectures. Our architectures, in contrast, have not been optimised to compare favourably against the state of the art, as our main objective is to develop a novel approach to multi-task learning leveraging synergies between label sets and knowledge of marginal distributions from unlabeled data. For example, we do not use pre-trained word embeddings (Augenstein et al., 2016; Palogiannidi et al., 2016; Vo and Zhang, 2015), class weighting to deal with label imbalance (Balikas and Amini, 2016), or domain-specific sentiment lexicons (Brun et al., 2016; Kumar et al., 2016). Nevertheless, our approach outperforms the state-of-the-art on two-way topic-based sentiment analysis (Topic-2).

The poor performance compared to the state-of-the-art on FNC and MultiNLI is expected; as we alternate among the tasks during training, our model only sees a comparatively small number of examples of both corpora, which are one and two orders of magnitude larger than the other datasets. For this reason, we do not achieve good performance on these tasks as main tasks, but they are still useful as auxiliary tasks as seen in Table 4.

6 Analysis

6.1 Label Embeddings

Our results above show that, indeed, modelling the similarity between tasks using label embeddings sometimes leads to much better performance. Figure 2 shows why. In Figure 2, we visualise the label embeddings of an MTL+LEL model trained on all tasks, using PCA. As we can see, similar labels are clustered together across tasks, e.g. there are two positive clusters (middle-right and top-right), two negative clusters (middle-left and bottom-left), and two neutral clusters (middle-top

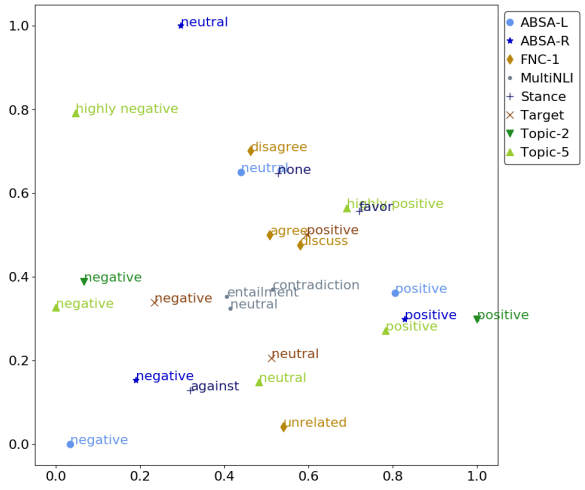


Figure 2: Label embeddings of all tasks. Positive, negative, and neutral labels are clustered together.

and middle-bottom).

Our visualisation also provides us with a picture of what auxilary tasks are beneficial, and to what extent we can expect synergies from multi-task learning. For instance, the notion of positive sentiment appears to be very similar across the topic-based and aspect-based tasks, while the conceptions of negative and neutral sentiment differ. In addition, we can see that the model has failed to learn a relationship between `MultiNLI` labels and those of other tasks, possibly accounting for its poor performance on the inference task. We did not evaluate the correlation between label embeddings and task performance, but Bjerva (2017) recently suggested that mutual information of target and auxiliary task label sets is a good predictor of gains from multi-task learning.

6.2 Auxilary Tasks

For each task, we show the auxiliary tasks that achieved the best performance on the development data in Table 4. In contrast to most existing work, we did not restrict ourselves to performing multi-task learning with only one auxiliary task (Søgaard and Goldberg, 2016; Bingel and Søgaard, 2017). Indeed we find that most often a combination of auxiliary tasks achieves the best performance. In-domain tasks are less used than we assumed; only `Target` is consistently used by all Twitter main tasks. In addition, tasks with a higher number of labels, e.g. `Topic-5` are used more often. Such tasks provide a more fine-grained reward signal, which may help in learning representations that generalise better. Finally, tasks with large amounts of training data such as `FNC-1` and `MultiNLI` are also used more often. Even if not directly related, the larger amount of training data that can be indirectly leveraged via multi-task learning may help the model focus on relevant parts of the representation space (Caruana, 1993). These observations shed additional light on when multi-task learning may be useful that go beyond existing studies (Bingel and Søgaard, 2017).

Main task	Auxiliary tasks
`Topic-2`	`FNC-1`, `MultiNLI`, `Target`
`Topic-5`	`FNC-1`, `MultiNLI`, `ABSA-L`, `Target`
`Target`	`FNC-1`, `MultiNLI`, `Topic-5`
`Stance`	`FNC-1`, `MultiNLI`, `Target`
`ABSA-L`	`Topic-5`
`ABSA-R`	`Topic-5`, `ABSA-L`, `Target`
`FNC-1`	`Stance`, `MultiNLI`, `Topic-5`, `ABSA-R`, `Target`
`MultiNLI`	`Topic-5`

Table 4: Best-performing auxiliary tasks for different main tasks.

6.3 Ablation analysis

We now perform a detailed ablation analysis of our model, the results of which are shown in Table 5. We ablate whether to use the LEL (+ *LEL*), whether to use the LTN (+ *LTN*), whether to use the LEL output or the main model output for prediction (main model output is indicated by , *main model*), and whether to use the LTN as a regulariser or for semi-supervised learning (semi-supervised learning is indicated by + *semi*). We further test whether to use diversity features (– *diversity feats*) and whether to use main model predictions for the LTN (+ *main model feats*).

Overall, the addition of the Label Embedding Layer improves the performance over regular MTL in almost all cases.

6.4 Label transfer network

To understand the performance of the LTN, we analyse learning curves of the relabelling function vs. the main model. Examples for all tasks without semi-supervised learning are shown in Figure 3. One can observe that the relabelling model does not take long to converge as it has fewer parameters than the main model. Once the relabelling model is learned alongside the main

	Stance	FNC	MultiNLI	Topic-2	Topic-5*	ABSA-L	ABSA-R	Target
MTL	44.12	72.75	49.39	**80.74**	0.859	74.94	82.25	65.73
MTL + LEL	**46.26**	72.71	**49.94**	80.52	0.814	74.94	79.90	**66.42**
MTL + LTN	40.95	72.72	44.14	78.31	0.851	73.98	82.37	63.71
MTL + LTN, main model	41.60	72.72	47.62	79.98	0.814	75.54	81.70	65.61
MTL + LEL + LTN	44.48	**72.76**	43.72	74.07	0.821	**75.66**	81.92	65.00
MTL + LEL + LTN, main model	43.16	72.73	48.75	73.90	0.810	75.06	**83.71**	66.10
MTL + LEL + LTN + main preds feats	42.78	72.72	45.41	66.30	0.835	73.86	81.81	65.08
MTL + LEL + LTN + main preds feats, main model	42.65	72.73	48.81	67.53	**0.803**	75.18	82.59	63.95
MTL + LEL + LTN + main preds feats – diversity feats	42.78	72.72	43.13	66.3	0.835	73.5	81.7	63.95
MTL + LEL + LTN + main preds feats – diversity feats, main model	42.47	72.74	47.84	67.53	0.807	74.82	82.14	65.11
MTL + LEL + LTN + semi	42.65	72.75	44.28	77.81	0.841	74.10	81.36	64.45
MTL + LEL + LTN + semi, main model	43.56	72.72	48.00	72.35	0.821	75.42	83.26	63.00

Table 5: Ablation results with task-specific evaluation metrics on test set with early stopping on dev set. *LTN* means the output of the relabelling function is shown, which does not use the task predictions, only predictions from other tasks. *LTN + main preds feats* means main model predictions are used as features for the relabelling function. *LTN, main model* means that the main model predictions of the model that trains a relabelling function are used. Note that for MultiNLI, we down-sample the training data. *: lower is better. Bold: best. Underlined: second-best.

Task	Main	LTN	Main (Semi)	LTN (Semi)
Stance	2.12	2.62	1.94	1.28
FNC	4.28	2.49	6.92	4.84
MultiNLI	1.5	1.95	1.94	1.28
Topic-2	6.45	4.44	5.87	5.59
Topic-5*	9.22	9.71	11.3	5.90
ABSA-L	3.79	2.52	9.06	6.63
ABSA-R	10.6	6.70	9.06	6.63
Target	26.3	14.6	20.1	15.7

Table 6: Error analysis of LTN with and without semi-supervised learning for all tasks. Metric shown: percentage of correct predictions only made by either the relabelling function or the main model, respectively, relative to the the number of all correct predictions.

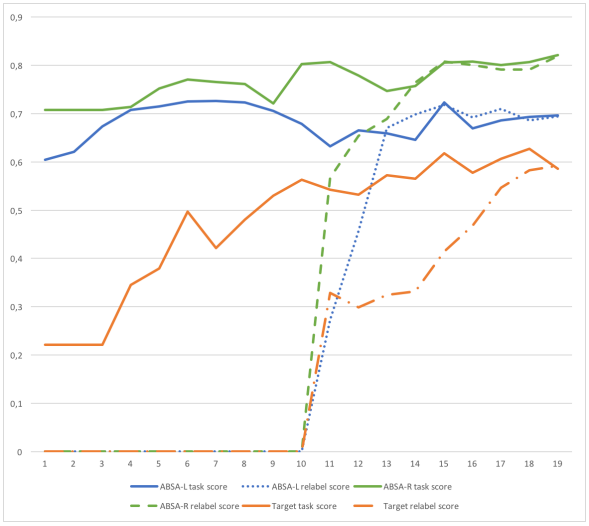


Figure 3: Learning curves with LTN for selected tasks, dev performances shown. The main model is pre-trained for 10 epochs, after which the relabelling function is trained.

model, the main model performance first stagnates, then starts to increase again. For some of the tasks, the main model ends up with a higher task score than the relabelling model. We hypothesise that the softmax predictions of other, even highly related tasks are less helpful for predicting main labels than the output layer of the main task model. At best, learning the relabelling model alongside the main model might act as a regulariser to the main model and thus improve the main model's performance over a baseline MTL model, as it is the case for TOPIC-5 (see Table 5).

To further analyse the performance of the LTN, we look into to what degree predictions of the main model and the relabelling model for individual instances are complementary to one another. Or, said differently, we measure the percentage of correct predictions made only by the relabelling model or made only by the main model, relative to the number of correct predictions overall. Results of this for each task are shown in Table 6 for the LTN with and without semi-supervised learning. One can observe that, even though the relabelling function overall contributes to the score to a lesser degree than the main model, a substantial number of correct predictions are made by the relabelling function that are missed by the main model. This is most prominently pronounced for ABSA-R, where the proportion is 14.6.

7 Conclusion

We have presented a multi-task learning architecture that (i) leverages potential synergies between classifier functions relating shared representations with disparate label spaces and (ii) enables learning from mixtures of labeled and unlabeled data. We have presented experiments with combinations of eight pairwise sequence classification tasks. Our results show that leveraging synergies between label spaces sometimes leads to big improvements, and we have presented a new state of the art for topic-based sentiment analysis. Our analysis further showed that (a) the learned label embeddings were indicative of gains from multi-task learning, (b) auxiliary tasks were often beneficial across domains, and (c) label embeddings almost always led to better performance. We also investigated the dynamics of the label transfer network we use for exploiting the synergies between disparate label spaces.

Acknowledgments

Sebastian Ruder is supported by the Irish Research Council Grant Number EBPPG/2014/30 and Science Foundation Ireland Grant Number SFI/12/RC/2289. Anders Søgaard is supported by the ERC Starting Grant Number 313695. Isabelle Augenstein is supported by Eurostars grant Number E10138. We further gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467* .

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016. Twitter Stance Detection with Bidirectional Conditional Encoding. In *Proceedings of EMNLP*.

Isabelle Augenstein and Anders Søgaard. 2017. Multi-task learning of keyphrase boundary detection. In *Proceedings of ACL*.

Georgios Balikas and Massih-Reza Amini. 2016. TwiSE at SemEval-2016 Task 4: Twitter Sentiment Classification. In *Proceedings of SemEval*.

Jonathan Baxter. 2000. A Model of Inductive Bias Learning. *JAIR* 12:149–198.

Joachim Bingel and Anders Søgaard. 2017. Identifying beneficial task relations for multi-task learning in deep neural networks. In *Proceedings of EACL*.

Johannes Bjerva. 2017. Will my auxiliary tagging task help? Estimating Auxiliary Tasks Effectivity in Multi-Task Learning. In *Proceedings of NODALIDA*.

Marcel Bollman, Joachim Bingel, and Anders Søgaard. 2017. Learning attention for historical text normalization by learning to pronounce. In *Proceedings of ACL*.

Caroline Brun, Julien Perez, and Claude Roux. 2016. XRCE at SemEval-2016 Task 5: Feedbacked Ensemble Modelling on Syntactico-Semantic Knowledge for Aspect Based Sentiment Analysis. *Proceedings of SemEval* .

Rich Caruana. 1993. Multitask Learning: A Knowledge-Based Source of Inductive Bias. In *Proceedings of ICML*.

Hongshen Chen, Yue Zhang, and Qun Liu. 2016. Neural Network for Heterogeneous Annotations. In *Proceedings of EMNLP*.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Recurrent neural network-based sentence encoder with gated attention for natural language inference. *arXiv preprint arXiv:1708.01353* .

Ronan Collobert and Jason Weston. 2008. A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In *Proceedings of ICML*.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research* 12:2493–2537.

Hal Daumé III. 2009. Bayesian multitask learning with latent hierarchies. In *Proceedings of UAI*.

Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive Recursive Neural Network for Target-dependent Twitter Sentiment Classification. In *Proceedings of ACL*. pages 49–54.

Ben Eisner, Tim Rocktäschel, Isabelle Augenstein, Matko Bosnjak, and Sebastian Riedel. 2016. emoji2vec: Learning Emoji Representations from their Description. In *Proceedings of SocialNLP*.

Theodoros Evgeniou, Charles A. Micchelli, and Massimiliano Pontil. 2005. Learning multiple tasks with kernel methods. *Journal of Machine Learning Research* 6:615–637.

Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Proceedings of EMNLP*.

Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. 2017. A Joint Many-Task Model: Growing a Neural Network for Multiple NLP Tasks. In *Proceedings of EMNLP*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the Knowledge in a Neural Network. *arXiv preprint arXiv:1503.02531* .

Laurent Jacob, Jean-Philippe Vert, Francis R Bach, and Jean-philippe Vert. 2009. Clustered Multi-Task Learning: A Convex Formulation. In *Proceedings of NIPS*. pages 745–752.

Robert a. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. 1991. Adaptive Mixtures of Local Experts. *Neural Computation* 3(1):79–87.

Zhuoliang Kang, Kristen Grauman, and Fei Sha. 2011. Learning with Whom to Share in Multi-task Feature Learning. In *Proceedings of ICML*.

Young-Bum Kim, Karl Stratos, Ruhi Sarikaya, and Minwoo Jeong. 2015. New Transfer Learning Techniques for Disparate Label Sets. In *Proceedings of ACL*.

Abhishek Kumar and Hal Daumé III. 2012. Learning Task Grouping and Overlap in Multi-task Learning. *Proceedings of the 29th International Conference on Machine Learning* pages 1383–1390.

Ayush Kumar, Sarah Kohail, Amit Kumar, Asif Ekbal, and Chris Biemann. 2016. IIT-TUDA at SemEval-2016 Task 5: Beyond Sentiment Lexicon: Combining Domain Dependency and Distributional Semantics Features for Aspect Based Sentiment Analysis. *Proceedings of SemEval* .

Ming Li and Zhi-Hua Zhou. 2007. Improve Computer-Aided Diagnosis With Machine Learning Techniques Using Undiagnosed Samples. *IEEE Transactions on Systems, Man and Cybernetics* 37(6):1088–1098.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial Multi-task Learning for Text Classification. In *Proceedings of ACL*.

Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task Sequence to Sequence Learning. In *Proceedings of ICLR*.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of SemEval*.

Preslav Nakov, Alan Ritter, Sara Rosenthal, Veselin Stoyanov, and Fabrizio Sebastiani. 2016. SemEval-2016 Task 4: Sentiment Analysis in Twitter. In *Proceedings of SemEval*. San Diego, California.

Nikita Nangia, Adina Williams, Angeliki Lazaridou, and Samuel R. Bowman. 2017. The RepEval 2017 Shared Task: Multi-Genre Natural Language Inference with Sentence Representations. In *Proceedings of RepEval*.

Elisavet Palogiannidi, Athanasia Kolovou, Fenia Christopoulou, Filippos Kokkinos, Elias Iosif, Nikolaos Malandrakis, Haris Papageorgiou, Shrikanth Narayanan, and Alexandros Potamianos. 2016. Tweester at SemEval-2016 Task 4: Sentiment Analysis in Twitter Using Semantic-Affective Model Adaptation. In *Proceedings of SemEval*. pages 155–163.

Hao Peng, Sam Thomson, Noah A Smith, and Paul G Allen. 2017. Deep Multitask Learning for Semantic Dependency Parsing. In *Proceedings of ACL 2017*.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual Part-of-Speech Tagging with Bidirectional Long Short-Term Memory Models and Auxiliary Loss. In *Proceedings of ACL*.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammed AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Veronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Núria Bel, Salud Maria Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 Task 5: Aspect Based Sentiment Analysis. In *Proceedings of SemEval*.

Marek Rei. 2017. Semi-supervised Multitask Learning for Sequence Labeling. In *Proceedings of ACL 2017*.

Benjamin Riedel, Isabelle Augenstein, Georgios P Spithourakis, and Sebastian Riedel. 2017. A simple but tough-to-beat baseline for the Fake News Challenge stance detection task. In *arXiv preprint arXiv:1707.03264*.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M. Marlin. 2013. Relation Extraction with Matrix Factorization and Universal Schemas. *Proceedings of NAACL-HLT* pages 74–84.

Sebastian Ruder, Joachim Bingel, Isabelle Augenstein, and Anders Søgaard. 2017. Sluice networks: Learning what to share between loosely related tasks. In *CoRR, abs/1705.08142*.

Sebastian Ruder, Parsa Ghaffari, and John G. Breslin. 2016. A Hierarchical Model of Reviews for Aspect-based Sentiment Analysis. *Proceedings of EMNLP* pages 999–1005.

Sebastian Ruder and Barbara Plank. 2017. Learning to select data for transfer learning with Bayesian Optimization. In *Proceedings of EMNLP*.

Anders Søgaard and Yoav Goldberg. 2016. Deep multi-task learning with low level tasks supervised at lower layers. In *Proceedings of ACL*.

Duy-Tin Vo and Yue Zhang. 2015. Target-Dependent Twitter Sentiment Classification with Rich Automatic Features. In *Proceedings of IJCAI*. pages 1347–1353.

Ya Xue, Xuejun Liao, Lawrence Carin, and Balaji Krishnapuram. 2007. Multi-Task Learning for Classification with Dirichlet Process Priors. *Journal of Machine Learning Research* 8:35–63.

Chih-Kuan Yeh, Wei-Chieh Wu, Wei-Jen Ko, and Yu-Chiang Frank Wang. 2017. Learning Deep Latent Space for Multi-Label Classification. In *Proceedings of AAAI*.

Kai Yu, Volker Tresp, and Anton Schwaighofer. 2005. Learning Gaussian processes from multiple tasks. *Proceedings of ICML* 22:1012–1019.

Yuan Zhang, Roi Reichart, Regina Barzilay, and Amir Globerson. 2012. Learning to Map into a Universal POS Tagset. In *Proceedings of EMNLP*.