Estimating the influence of auxiliary tasks for multi-task learning of sequence tagging tasks

Fynn Schröder
Language Technology Group
Universität Hamburg
Hamburg, Germany
fschroeder@informatik.uni-hamburg.de

Chris Biemann
Language Technology Group
Universität Hamburg
Hamburg, Germany
biemann@informatik.uni-hamburg.de

Abstract

Multi-task learning (MTL) and transfer learning (TL) are techniques to overcome the issue of data scarcity when training state-of-the-art neural networks. However, finding beneficial auxiliary datasets for MTL or TL is a time- and resource-consuming trial-and-error approach. We propose new methods to automatically assess the similarity of sequence tagging datasets to identify beneficial auxiliary data for MTL or TL setups. Our methods can compute the similarity between any two sequence tagging datasets, i.e. they do not need to be annotated with the same tagset or multiple labels in parallel. Additionally, our methods take tokens and their labels into account, which is more robust than only using either of them as an information source, as conducted in prior work. We empirically show that our similarity measures correlate with the change in test score of neural networks that use the auxiliary dataset for MTL to increase the main task performance. We provide an efficient, open-source implementation.[1]

1 Introduction

State-of-the-art neural networks usually require large amounts of training data and vast computational resources. Especially for low-resource tasks, data scarcity is the main issue hampering the training of robust models. By leveraging multi-task learning or transfer learning, auxiliary data can be incorporated into the training to boost the main task performance. Finding suitable auxiliary datasets for these cases is a time- and resource-consuming trial-and-error approach, because there can be plenty of plausible auxiliary datasets that could help to learn the main task. For a proper evaluation of different auxiliary datasets, hyperparameter search and training runs with multiple random seeds have to be performed for each auxiliary dataset individually. Thus, the process takes even longer and uses even more computational resources. We propose methods to shorten this trial-and-error approach by computing the similarity between any two sequence tagging datasets. Based on the similarity, suitable datasets can be quickly selected to be used as auxiliary training data for multi-task or transfer learning.

Our contributions are a family of novel methods to compute the similarity of sequence tagging datasets, where the similarity values correlate with the change in multi-task learning performance when using one dataset as auxiliary data for training the other. We evaluate our methods in experiments with five part-of-speech (POS) tagging, nine named-entity recognition (NER) and three argumentation mining (AM) datasets. Our similarity measures allow for comparison both datasets for the same and different tasks, not requiring the same set of labels on target and auxiliary dataset. The calculated similarity scores can be used to predict which dataset will be beneficial as auxiliary training data for multi-task training in order to shorten the search process.

2 Related work

2.1 Neural multi-task and transfer learning

Multi-task learning (MTL) is a technique to learn multiple tasks jointly (Caruana, 1997). Depending on the setting, either all tasks are equally important, or only the performance on the main task is of interest, which shall be improved with additional training data. MTL has been successfully applied in natural language processing for various sequence tagging tasks (Søgaard and Goldberg, 2016; Bjerva et al., 2016; Plank et al., 2016; Martínez Alonso and Plank, 2017; Kaiser et al., 2017; Bingel and Søgaard, 2017; Augenstein and Søgaard, 2017; Kim et al., 2017; Yang et al., 2017; Changpinyo

[1] github.com/uhh-lt/seq-tag-sim

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 2971–2985
July 5 - 10, 2020.

et al., 2018; Liu et al., 2018; Schulz et al., 2018). These approaches use hard parameter sharing in the hidden layers of neural learning architectures, where the same weights are updated from several tasks. The majority of works combined a main task with a single, supervised auxiliary task.

In transfer learning, a model is pre-trained on an auxiliary dataset to increase the main task performance. Howard and Ruder (2018) showed knowledge transfer based on large-scale language modeling. Before the breakthrough with BERT (Devlin et al., 2019), only partial knowledge transfer via word embeddings such as word2vec (Mikolov et al., 2013) or ELMo (Ilić et al., 2018) was utilized.

2.2 Effect of auxiliary task similarity

In theory, auxiliary tasks can have various relationships to the main task (Ruder, 2017). In practice, the most common choice is to use a "somehow" related task. Caruana (1997) argues that tasks are similar if the same features are used for making predictions. Baxter (2000) suggests similar tasks should have the same inductive bias. Ben-David and Schuller (2003) indicate that tasks originating from the same probability distribution are similar and perform well in an MTL setting. No universal measure for task similarity exists, but it is needed to select tasks to prefer for training (Ruder, 2017).

Although MTL is frequently applied in recent work, few elaborate on the effect of task and dataset similarity. Recent work on neural MTL found different hints regarding task similarity that are only applicable to a specific scenario. Kim et al. (2017) performed MTL on POS tagging across 14 languages and found that language similarity seems to correlate with MTL performance. Yang et al. (2017) worked on common tasks with artificially reduced datasets. They attribute the degree of performance increase to label abundance for the main task, dataset similarity and number of shared parameters. Changpinyo et al. (2018) compared eleven tasks and observed that some tasks increase the performance in most cases, while tasks with a small tagset decreased the main task performance. In contrast, Martínez Alonso and Plank (2017) show results that auxiliary tasks with few labels and a uniform label distribution perform better for MTL in neural sequence tagging: Auxiliary tasks having many labels or high entropy harm the main task performance. While Ruder et al. (2019) confirm these findings, Bjerva (2017) found no evidence of label entropy correlating with MTL performance. Martínez Alonso and Plank (2017) found a difference between two POS datasets when used as auxiliary data because converting one to another tagset changes the effect of MTL significantly.

Kim et al. (2015) propose a method using label embeddings to map labels from auxiliary datasets to the target tagset so that MTL can be treated as single-task learning (STL) with an increased amount of training data. Bingel and Søgaard (2017) predict MTL performance from dataset and STL learning features and found the learning curve to be much more important. From the dataset features, the number of labels on the main task and the auxiliary label entropy showed predictive potential.

Most similar to our approach is the work of Bjerva (2017), who estimates the effect of an auxiliary task in MTL with information-theoretic measures. As the method requires the same datasets to be tagged with multiple tasks in parallel, at least one task must be automatically taggable with almost perfect results. He shows a correlation of conditional entropy and mutual information with a change in accuracy compared to STL. Results on the semantic task of Bjerva et al. (2016); Martínez Alonso and Plank (2017) indicate that mutual information for helpful auxiliary tasks is higher than for harmful tasks.

Augenstein et al. (2018) propose an architecture that learns label embeddings for natural language classification tasks and find that label embeddings indicate gains or harms of MTL. Ruder et al. (2019) correlate task properties with performance differences and learned meta-network parameters of their proposed sluice networks. They find that MTL gains are higher for smaller training datasets and that sluice networks learn to share more in case of higher variance in the training data.

Opposed to previous approaches, our methods can compare same-task datasets and are not restricted to datasets with parallel labels. As our experiments in Section 5 require these properties, previous approaches are not applicable and thus not comparable. Next, we will introduce information-theoretic measures that build the foundation for our dataset similarity measures proposed in Section 4.

3 Information-theoretic clustering comparison measures

Entropy is a measure of the uncertainty of a random variable. The entropy $H(X)$ of a discrete random

variable X with alphabet $\mathcal{X}$ is defined as

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log_2 p(x) \quad (1)$$

where $p(x)$ is the probability mass function $p(x) = \Pr\{X = x\}, x \in \mathcal{X}$. It is 0 when $p = 0$ or 1 and maximal when $p = \frac{1}{|\mathcal{X}|}$ (uniform distribution) with an upper bound of $H(X) \leq \log_2 |X|$.

Joint entropy $H(X, Y)$ extends entropy from a single to two random variables. For a pair of discrete random variables (X, Y) with a joint probability distribution $p(x, y)$, it is defined as

$$H(X, Y) = -\sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log_2 p(x, y). \quad (2)$$

Mutual information (MI) $I(X; Y)$ describes the amount of information one random variable X contains about another Y. It is a symmetric measure of range $[0, \min\{H(X), H(Y)\}]$ defined as

$$I(X; Y) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log_2 \frac{p(x, y)}{p(x)p(y)} \quad (3)$$

with probability mass functions $p(x)$, $p(y)$ and a joint probability mass function $p(x, y)$. For a detailed description of entropy, mutual information and information theory in general, please refer to Cover and Thomas (2006).

A clustering C is a way to partition a dataset D into non-overlapping subsets $\{c_1, c_2, \dots\}$ together containing all N items of D. Comparing clusterings requires a measure to determine the quality of a clustering according to another clustering, e.g. the ground truth. Such a measure should quantify the amount of information shared between both clusterings. (Vinh et al., 2010)

Information-theoretic clustering comparison measures are based on a solid mathematical foundation from information theory and can work with non-linear similarities. They have become popular by the works of Strehl and Ghosh (2003) and Meilă (2005).

Mutual information measures the information shared between two clusterings C and C'. A higher MI signals a greater help in predicting the cluster labels in C with information from C'. Several normalized mutual information variants can be derived:

$$NMI_{joint} = \frac{I(C; C')}{H(C, C')} \quad (4)$$

$$NMI_{max} = \frac{I(C; C')}{\max(H(C), H(C'))} \quad (5)$$

Analogously to NMI_{max}, there are NMI_{sum}, NMI_{sqrt} and NMI_{min} that use entropy sums, square root of the entropy products or minimum of both entropy values as a normalization factor (Kvalseth, 1987; Strehl and Ghosh, 2003; Yao, 2003; Liu et al., 2008). They are all bounded in $[0, 1]$, equaling 0 when two clusterings share no information at all, i.e. are fully independent and 1 when two clusterings are identical.

According to Vinh et al. (2010), NMI_{max} and NMI_{joint} satisfy the highest number of theoretical properties desirable among the clustering comparison measures. They prove that only the unit complements of both measures satisfy the *metric property* (*positive definiteness*, *symmetry* and *triangle inequality*). While all measures satisfy the *normalization property*, none conform to the *constant baseline property* unless the number of items N is large, compared to the number of clusters.

4 Method

The high-level idea of our dataset similarity measures is the following: Words and labels from one dataset are correlated with the words and their labels from another dataset to create a probabilistic mapping between both label sets. Either an exact string matching or a fuzzy matching based on word embedding representations can be used. The dataset similarity is measured via the quality of this label mapping.

4.1 Casting label similarity as a clustering comparison problem

Transforming the problem of token-label dataset similarity to a clustering comparison problem allows reusing existing clustering comparison measures. A clustering represents one label set, and each label is a cluster within the clustering, i.e. all tokens having the same label belong to one cluster.

A contingency table, also called a confusion matrix, is a handy tool to compare clusterings. Let us assume that a dataset D is annotated with two labels in parallel from two tasks T and T' with arbitrary label sets L and L'. The comparison of L with L' on D can be transformed into a clustering comparison problem. The clusters for T are the labels $l_1, l_2, \dots, l_N$ when the label set L has N different labels in total. The clusters for T' are labeled analogously $l'_1, l'_2, \dots, l'_M$ for the M labels in the set L'. Table 1 shows the resulting contingency table for the described setting. The values c_{xy} are

the counts how many tokens are in the dataset that are labeled as / belong to cluster l_x in task T and simultaneously l'_y in the task T'.[2]

	l'_1	l'_2	...	l'_M	Σ
l_1	c_{11}	c_{12}	...	c_{1M}	$c_{1.}$
l_2	c_{21}	c_{22}	...	c_{2M}	$c_{2.}$
...	...	...		...	...
l_N	c_{N1}	c_{N2}	...	c_{NM}	$c_{N.}$
Σ	$c_{.1}$	$c_{.2}$	...	$c_{.M}$	c

Table 1: Contingency table for a comparison of label sets L and L' with N and M unique labels

Based on the counts in the contingency table, information-theoretic measures such as (joint) entropy or mutual information can be calculated. Because the probability mass functions $p(x)$, $p(y)$ and $p(x, y)$ are unknown for the label sets L and L' in dataset D, the probabilities are approximated by the relative frequencies of the label pairs. The entropy of both label sets has to be taken into account to know whether the tasks T and T' are similar, i.e. a normalized mutual information variant shown in Equations 4 and 5 has to be used. With the notation in Table 1, the NMI_{joint} definition becomes

$$NMI(L, L')_{joint} = \frac{I(L; L')}{H(L, L')} = \frac{\sum_{i=1}^{N}\sum_{j=1}^{M}\frac{c_{ij}}{c}\log_2\left(\frac{c_{ij}c}{c_{i.}c_{.j}}\right)}{-\sum_{i=1}^{N}\sum_{j=1}^{M}\frac{c_{ij}}{c}\log_2\left(\frac{c_{ij}}{c}\right)}. \quad (6)$$

The other measures can be changed analogously.

Next, we show how to transform label similarity to clustering comparison without being restricted to datasets annotated in parallel with both label sets.

4.2 Obtaining label pairs from datasets

To compare two datasets, one of the datasets can be tagged automatically with the other task's labels as proposed by Bjerva (2017). However, a comparison is only possible if at least one of the tasks can be tagged automatically with near-perfect accuracy. While the necessary performance-level has been reached for a few simple tasks, the state-of-the-art performance on most tasks seems insufficient for this purpose. Further, two datasets of the same task, e.g. two NER datasets with the same tagset, cannot be meaningfully compared when tagged automatically. We propose two approaches to lift the restrictions on the datasets and tasks. The solutions enable a comparison of arbitrary task and dataset combinations.

4.2.1 Text overlap

If a manually defined one-to-one mapping from labels of one dataset to another one exists, datasets can be compared to each other using this label mapping function, because it produces a dataset with parallel label sets. While mapping a fine-grained label set to a coarse label set is possible, it is unclear how to map a coarse label to finer sub-labels.

The *text overlap* approach implicitly generates a label mapping from the token-label pairs of both datasets. This has the advantage of being independent of external knowledge and enabling a probabilistic mapping from coarse to fine-grained label sets specific to the datasets. Tokens are aggregated so that a token is associated with the number of times it has been tagged with each label. Only tokens occurring in both datasets can be used to fill in the counts of a contingency table. By looking only at the intersection of tokens occurring in both datasets, a new virtual dataset is created, where each token is tagged with two labels. For each token, the count at the position (l_i, l'_j) in the contingency table is increased by a combination of the number of times the current token was tagged with labels l_i and l'_j. With the *additive* method to fill a contingency table, label counts for words from both datasets are added because they are viewed as multiple instances from one dataset.[3]

An alternative to addition is to use multiplication to combine the counts for matching words. The counts for each label combination are multiplied and added at the corresponding position in the contingency table. An effect of this approach is that words being frequent in both datasets contribute more to be counts. There are more possible schemes on how to combine the raw counts from two datasets into a mutual contingency table. Similarity measures such as NMI can be computed on any contingency table obtained from these methods.

An advantage of the text overlap approach is that it is fast because it only involves text processing and a few counts. The downside is that an identical dataset can only be identified with 100% similarity if each word always has the same label. Another issue is that only a fraction of each dataset is used

[2] Illustrating examples are provided in Appendix A.1

[3] Illustrating examples are provided in Appendix A.2

for the actual comparison. As the plain text overlap approach does not consider the ratio of shared vocabulary, it is possible to have a "false positive", i.e. a high similarity is reported for two datasets although they share only one word. To fix this, we combine the NMI value and the ratio of shared vocabulary (SV) via the harmonic mean into our text overlap (TO) measure

$$TO = \frac{2 \cdot NMI \cdot SV}{NMI + SV} \tag{7}$$

with the shared vocabulary

$$SV = \frac{|V \cap V'|}{|V \cup V'|} \tag{8}$$

where V and V' are the sets of all unique words in the two datasets D and D'.

When constructing the contingency table (e.g. Table 1) with the text overlap approach, the sequence information of label-word pairs, i.e. the context, cannot be captured in the counts. With the usage of contextual embeddings, this issue can be mitigated sufficiently.

4.2.2 Vector space similarity

Word embeddings allow representing words in the form of dense vectors within a vector space instead of a specific character sequence in the language's vocabulary. Thus, it is possible to perform mathematical operations on these vectors and compute e.g. the semantic similarity of two words by computing their cosine similarity within the vector space (Elekes et al., 2017). These word vector techniques can be used to tackle the problems of the previously shown text overlap approach.

A first extension allows incorporating words not occurring in both datasets. Vector representations are obtained for each unique word in the datasets. Instead of ignoring words contained only in one dataset, the closest word from the other dataset is chosen via cosine similarity for the pairwise label comparison. The remaining process and similarity measure computation stays the same.[4]

In the vector space approach, all tokens are compared. For each token, a unique vector representation is obtained via contextual embeddings such as *ELMo* (Ilić et al., 2018) or *BERT* (Devlin et al., 2019). In order to fill in the counts of a contingency table, each token from one dataset is matched with the most similar vector representation in the other dataset and the count for the label-pair is increased by the vector space similarity of the two tokens.[4] The usage of contextual embeddings allows to incorporate the sequence information of label-word pairs into the counts. A similarity measure like NMI can be calculated from these counts as before. Identical datasets can be scored with 100% similarity when the contextual embeddings are able to produce unique vector representations for each token. In general, this method handles ambiguity in language much better as compared to the plain text approach, which should help to improve the similarity comparison between various datasets. Because the process of selecting the closest vector representation from the main dataset to the auxiliary dataset or vice versa can result in different combinations, the counts in the contingency table will be different depending on the direction. Thus, for a symmetric similarity measure like NMI, two scores are obtained. We further combine the forward and backward direction using the harmonic mean into a unified undirectional embedding (UUE) measure:

$$UUE = \frac{2 \cdot NMI_{forward} \cdot NMI_{backward}}{NMI_{forward} + NMI_{backward}} \tag{9}$$

The forward and backward NMI in Equation 9 use the same NMI formula and applies it to different counts obtained from the two directions of embeddings comparisons. In our experiments, the actual NMI formula is either NMI_{max} or NMI_{joint} due to their desirable theoretical properties.

5 Experiments

In this section, experiments will be performed to check whether the similarity of two datasets correlates with the effect on the MTL performance when using the second dataset as auxiliary training data.

5.1 Controlled environment experiments

Before the similarity measures are evaluated together with the MTL performance, we evaluate them independently in a controlled environment. We perform a sanity check by comparing the similarity scores with the intuitive, expected outcome.

Two POS tagging datasets (WSJ, EWT) and two NER datasets (CNLE, ONT) shown in Table 2 will be used to sample three new, non-overlapping datasets each. The samples are named e.g. WSJ-1, WSJ-2, and WSJ-3. Their sizes are equal to ⅙, ⅔ and ⅚ of the original number of tokens. Under the assumption that the similarity within samples from

[4] Illustrating examples are provided in Appendix A.3

the same original dataset is higher than the similarity between samples from different datasets, the pairwise NMI scores can be qualitatively evaluated.

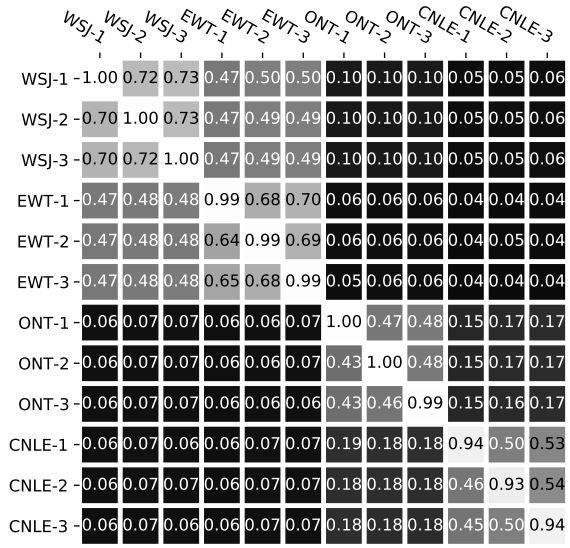

Figure 1: Pairwise NMI_{joint} similarity scores (Equation 6) obtained on contingency tables filled with the vector space similarity approach using contextual BERT embeddings. The heat map encodes the values from 0.0 in black to 1.0 in white.

Figure 1 shows the pairwise NMI_{joint} similarity scores obtained with Equation 6 between these twelve samples. The pairs of identical datasets create a visible diagonal line of maximal similarity. The visible 3×3 blocks along the diagonal show high similarity scores and are aligned with comparisons of samples within the same original dataset. Per row or column, the values within these blocks are higher than any other value outside. Thus, the NMI_{joint} score allows identifying other samples of the same original datasets.

Another interesting property is that the similarity between samples of the two original POS tagging datasets (WSJ, EWT) is higher than the similarity between any POS–NER pair. The same is true the other way around for the NER dataset samples (CNLE, ONT). Hence, the NMI_{joint} score can be used to distinguish datasets of the same task from others. Note that all four original datasets use different tagsets with a greatly varying number of tags (see Table 2) and that neither the shared vocabulary nor the joint label entropy can be employed to distinguish the POS and NER samples correctly.[5]

Overall, the NMI_{joint} scores presented in Figure 1 agree with the intuition which dataset samples should be similar. For each row or column, the similarity values can be ordered descending by identical, same original dataset, same task, and other samples.

5.2 Experimental setup

Experiments to correlate dataset similarity and the network's multi-task learning performance will be performed a) using two neural network architectures with Softmax and conditional random field classifiers, b) for the tasks of POS tagging, NER, and AM, c) on multiple datasets per task. Table 2 shows the datasets used in the experiments. Similar to Yang et al. (2017), we sample new training datasets as subsets of the originals to show a larger influence of auxiliary data as there is no room for improvement for simple tasks on large training sets. For the auxiliary datasets, subsets of different sizes are sampled to allow a fair comparison of the performance effect. The standard development and test sets of the original datasets are used if available. Otherwise, random samples without overlap with any other subsampled dataset are used.

From the POS tagging datasets, a new training dataset of 25 000 tokens is sampled for WSJ, BC, and EWT. From all POS tagging datasets, auxiliary datasets of increasing size are sampled containing $25, 50, 100, 250, 500, 1000 \times 1000$ tokens limited by the size of the original dataset.

For NER, training sets of 50 000 tokens are sampled from all datasets except GMB, SEC, and WNUT. Auxiliary datasets containing $50, 100, 250 \times 1000$ tokens are created for all datasets whenever possible.

For AM, we use the full PE and WD datasets for training and as auxiliary data. We sample auxiliary data from the IBM data equal in size to the others.

As the primary concern of the experiments is to enable significant differences in the neural network results with different auxiliary datasets, the network shares most of its parameters. In order to allow every training and auxiliary dataset combination to use their full potential, all relevant hyperparameters are tested for each pair of training and auxiliary dataset similar to Schulz et al. (2018).

The neural network architecture for the experiments uses hard parameter sharing with a bidirectional gated recurrent unit (GRU) (Cho et al., 2014), a simpler version of the long short-term memory (Hochreiter and Schmidhuber, 1997), that is commonly used in MTL sequence tagging works

[5] See Figures 3 and 4 in Appendix A.4 for details.

ID	Dataset	Reference	Tokens	Tags	STL performance
		PART-OF-SPEECH TAGGING DATASETS			
BNC	British National Corpus	BNC Consortium (2007)	111 973 625	91	-
WSJ	Penn Treebank Wall Street Journal	Marcus et al. (1999)	1 286 980	45	86.35 ± 0.26
BC	Penn Treebank Brown Corpus	Marcus et al. (1999)	1 162 358	45	85.61 ± 0.35
EWT	UD English Web Treebank	Silveira et al. (2014)	254 854	17	88.35 ± 0.42
GSD	UD German GSD	McDonald et al. (2013)	297 836	17	-
		NAMED-ENTITY RECOGNITION DATASETS			
ONT	English OntoNotes Release 5.0	Weischedel et al. (2013)	2 001 102	37	47.53 ± 0.83
CNLE	CoNLL'03 Shared Task (English)	Tjong Kim Sang and De Meulder (2003)	301 418	9	70.30 ± 2.50
CNLG	CoNLL'03 Shared Task (German)	Tjong Kim Sang and De Meulder (2003)	310 318	9	41.62 ± 0.27
EPG	Part of EUROPARL (German)	Faruqui and Padó (2010)	110 405	9	86.99 ± 0.42
GEN	GermEval 2014 NER Shared Task	Benikova et al. (2014)	591 005	24	26.97 ± 1.16
GMB	Groningen Meaning Bank 2.2.0	Bos et al. (2017)	1 354 149	17	-
SEC	SEC filings	Salinas Alvarado et al. (2015)	54 256	8	-
WIKI	Wikigold	Balasuriya et al. (2009)	39 152	8	67.19 ± 1.38
WNUT	W-NUT'17 Shared Task	Derczynski et al. (2017)	101 736	13	-
		ARGUMENTATION MINING DATASETS			
PE	Persuasive Essays (version 2)	Stab and Gurevych (2017)	148 182	11	53.71 ± 1.01
WD	Web Discourse	Habernal and Gurevych (2017)	84 817	12	24.58 ± 1.32
IBM	IBM Debater	Levy et al. (2018)	48 626 006	5	-

Table 2: Datasets used to sample new training or auxiliary datasets. The number of tags is a generic count, where e.g. `B-PER` and `I-PER` are considered to be different tags. STL performance (accuracy for POS, else macro F1 score) is not obtained on the full, but on the sampled training sets. STL scores are not shown for datasets only used as auxiliary data. Note that the IBM dataset contains many duplicate claims and near-duplicate sentences.

(see Section 2.1). Apart from self-learned word embeddings, character features based on another bidirectional GRU are included. Similar to Plank et al. (2016); Martínez Alonso and Plank (2017); Bjerva (2017); Ruder et al. (2019) we decided against pre-trained word embeddings in the network to avoid any influence on the comparison of STL and MTL performance. The last two, task-specific layers transform the GRU's hidden state to the task-specific labels and apply either a Softmax or conditional random field (CRF) (Lafferty et al., 2001) to predict the label.[6]

Auxiliary data is only used for the same task, i.e. no POS tagging dataset is used as auxiliary training data for NER and vice versa. For POS tagging, 81 pairs of training and auxiliary datasets are tested with 64 hyperparameter combinations and three random seeds. In the case of NER, 117 pairs of training and auxiliary datasets are tested with two neural network models, 16 hyperparameter combinations, and three random seeds. In total, 26 784 training runs have been performed.

We compute the similarities for pairs of training and auxiliary datasets in three ways. The text overlap approach is used with and without word embeddings. For the latter, 300-dimensional fastText embeddings[7] with sub-word information are used that consist of 2 million word vectors trained on the Common Crawl (Mikolov et al., 2018). We evaluate the additive and multiplicative ways with multiple weighting schemes to combine the label counts and calculate various similarity measures from the resulting contingency table. The "BERT-Base Multilingual Cased" model (Devlin et al., 2019) is used for the third, token-based approach.

5.3 Results and analysis

In Figure 2, the difference in accuracy over STL is plotted against the UUE NMI_{joint} similarity measure using BERT embeddings. Overall, the data points are scattered from the bottom left to the top right. There are no cases of low similarity coinciding with high accuracy increase. The data points with auxiliary data from the German GSD dataset are clustered close to the bottom left, i.e. low similarity and almost no accuracy gain. This concurs with the intuition that using a German auxiliary dataset for an English training dataset should not lead to a significant performance increase. The data points with auxiliary data from the same original dataset as the training set are clustered to the top right, i.e. have the highest similarity and performance increase as expected. The scatter plots for

[6] Training procedure and hyperparameters are described in more detail in Appendix A.5

[7] `crawl300d2Msubword.zip` from `fasttext.cc`

other sizes of auxiliary data and methods, e.g. computing NMI_{max} on the contingency table from the text overlap approach, look similar.

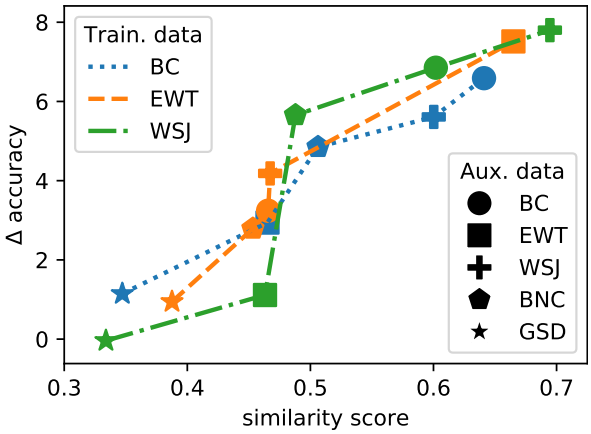

Figure 2: Plot comparing the POS tagging difference in accuracy between STL and MTL (auxiliary size 250 000 tokens) with the UUE NMI_{joint} similarity obtained using BERT embeddings for each token

To quantify the various similarity computation methods, we correlate the change in accuracy with the similarity value. Table 3 shows the median and mean correlation of similarity with change in accuracy for the best ten methods averaged over groups of identically-sized auxiliary datasets. As a baseline, the correlation with the ratio of shared vocabulary is included. We only show the results for NMI_{joint} as the correlation was equal to or better than NMI_{max} in most cases. The correlation between the similarity and change in accuracy is strong according to both Kendall's rank correlation and Pearson's linear correlation coefficients, which is in line with the plot shown in Figure 2. Since the p-values for the similarity methods are well below 0.005, it is *very* unlikely that similarity and accuracy are not correlated. The strongest correlation, according to Kendall's τ, is achieved with the harmonic mean of shared vocabulary and multiplicative text overlap. According to Pearson's ρ, the highest linear correlation is achieved with the UUE (Equation 9) vector space method, which is depicted in Figure 2. The correlation coefficients of the text overlap approach are consistently higher than the shared vocabulary baseline since the baseline is oblivious to the labels.

For NER, the results are shown in Table 4. In comparison to the POS tagging results, methods using embeddings perform better than those without. The strongest Kendall and Pearson correlations are achieved by the vector space approach computing the joint NMI on a contingency table filled from forward BERT embeddings. While a linear correlation on the POS tagging results was deemed reasonable based on a data analysis, the Pearson correlation values for NER might be prone to outlier effects and are therefore only included for completeness.

For AM, no quantitative analysis could be performed due to a limited number of samples. With MTL, the performance on PE increased to 54.26 when using WD as auxiliary data, while IBM reduced it to 51.37. WD performance is slightly reduced by PE as auxiliary data to 21.72, but reduced to 9.42 by IBM. While we saw no correlation with the text overlap similarities, the forward vector space measure matches the MTL score change

Primary method	Combination	Count method	Embedding	$\tilde{\tau}$	Kendall's $\bar{\tau}$	$\tilde{\rho}$	Pearson's $\bar{\rho}$
text overlap & SV	TO	multiplicative	-	0.73	0.71 ± 0.05	0.80	0.79 ± 0.07
text overlap & SV	TO	additive	-	0.72	0.72 ± 0.10	0.78	0.79 ± 0.04
text overlap	-	multiplicative	fastText	0.70	0.69 ± 0.08	0.83	0.82 ± 0.07
vector space	UUE	-	BERT	0.70	0.69 ± 0.12	0.84	0.84 ± 0.06
vector space	-	-	BERT	0.69	0.65 ± 0.09	0.83	0.82 ± 0.06
text overlap	-	multiplicative	-	0.68	0.64 ± 0.12	0.73	0.74 ± 0.08
text overlap	UUE	additive	-	0.67	0.66 ± 0.12	0.75	0.77 ± 0.06
text overlap	-	additive	-	0.67	0.65 ± 0.11	0.74	0.76 ± 0.06
text overlap	-	additive	-	0.66	0.64 ± 0.12	0.68	0.69 ± 0.08
text overlap	UUE	multiplicative	fastText	0.65	0.65 ± 0.11	0.83	0.83 ± 0.04
shared vocabulary	-	-	-	0.63	0.60 ± 0.14	0.77	0.75 ± 0.07

Table 3: Correlation between various NMI_{joint} similarity measures and the change in POS tagging accuracy using MTL. The entries show the median and mean of Kendall's and Pearson's correlation coefficients sorted descendingly by $\tilde{\tau}$. The average p-values for all methods (except the shared vocabulary baseline) are below 0.005.

Primary method	Combination	Count method	Embedding	$\tilde{\tau}$	Kendall's $\bar{\tau}$	$\tilde{\rho}$	Pearson's $\bar{\rho}$
vector space	-	-	BERT	0.65	0.62 ± 0.06	0.95	0.92 ± 0.05
vector space	UUE	-	BERT	0.59	0.55 ± 0.11	0.89	0.89 ± 0.05
text overlap	-	multiplicative	fastText	0.57	0.54 ± 0.09	0.91	0.88 ± 0.07
text overlap	-	additive	fastText	0.57	0.54 ± 0.09	0.87	0.86 ± 0.05
text overlap	UUE	multiplicative	fastText	0.52	0.50 ± 0.13	0.80	0.83 ± 0.06
text overlap & SV	TO	additive	-	0.51	0.50 ± 0.13	0.81	0.79 ± 0.04
text overlap & SV	TO	multiplicative	-	0.51	0.50 ± 0.13	0.80	0.79 ± 0.06
text overlap	UUE	additive	fastText	0.49	0.48 ± 0.08	0.83	0.84 ± 0.04
text overlap	-	multiplicative	-	0.47	0.44 ± 0.11	0.83	0.82 ± 0.08
text overlap	-	additive	-	0.42	0.41 ± 0.07	0.82	0.80 ± 0.04
shared vocabulary	-	-	-	0.48	0.49 ± 0.13	0.75	0.73 ± 0.05

Table 4: Correlation between NMI_{joint} various similarity measures and the change in NER F1 score using MTL. The entries show the median and mean of Kendall's and Pearson's correlation coefficients sorted descendingly by $\tilde{\tau}$. The average p-values for all methods (except the shared vocabulary baseline) are below 0.001. The change in F1 score was highly affected by random initialization, so the correlation scores must be used with caution.

when comparing averaged span embeddings: The NMI_{joint} similarity of PE–IBM is 0.09, and PE-WD is measured 0.26 whereas WD–PE has a similarity score of 0.06 and WD–IBM is scored 0.04. Thus, our similarity measure identifies the most promising auxiliary dataset also in this case.

Overall, there is a strong correlation between MTL scores and dataset similarity computed by our proposed methods. In the case of POS tagging, the correlation is impressive — it is visible in the scatter plot and accompanied by high-confidence correlation coefficients. The results for NER are less clear but still indicate that similarity and test set performance are correlated.

We can recommend the text overlap approach combined with the shared vocabulary for syntactic tasks with single-token labels. It performed the best in our POS tagging evaluation and is computed in less than a second. Both additive and multiplicative count combination methods worked equally well in our tests. For more complex tasks such as NER or AM and in case labels span multiple tokens, we suggest using the approach based on the forward vector space similarity. It performed the best in our NER evaluation. Further, it was the only method to work reasonably well with the AM datasets because spans of multiple tokens could be compared by combining the embeddings of all contained tokens. In all cases, we recommend using the mutual information normalized by the joint entropy NMI_{joint} as the actual similarity measure because it was either equal to or better than the other variants.

6 Conclusion

The similarity measures allow distinguishing good from bad candidates for usage as auxiliary data. This is an *immensely* valuable information as the number of expensive neural network training runs can be reduced to a fraction while still finding the best auxiliary dataset(s) to increase performance on the main task. In contrast to previous methods, our measures do not require the label sets to be the same and do not require automatic tagging. The experiments show that similarity measures allow ordering the effects of auxiliary datasets by direction and intensity for an individual training dataset. Our experimental findings are also supported from a theoretical point of view. The developed methods working on both words and their labels have a substantial advantage over approaches that are based only on words or the label distributions. The quick similarity calculation can improve the main task performance when better datasets are used as auxiliary data that would never have made it through the otherwise purely manual preselection process.

In future work, apart from improving the similarity measures, it could be examined to predict MTL scores or estimate the right amount of auxiliary data or shared parameters in the neural network.

Acknowledgments

We would like to thank all anonymous reviewers for their valuable feedback. This work was partially funded by the Cluster of Excellence CLICCS (EXC 2037), Universität Hamburg, funded through the German Research Foundation (DFG).

References

Isabelle Augenstein, Sebastian Ruder, and Anders Søgaard. 2018. Multi-task learning of pairwise sequence classification tasks over disparate label spaces. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1896–1906, New Orleans, Louisiana. Association for Computational Linguistics.

Isabelle Augenstein and Anders Søgaard. 2017. Multi-task learning of keyphrase boundary classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 341–346, Vancouver, Canada. Association for Computational Linguistics.

Dominic Balasuriya, Nicky Ringland, Joel Nothman, Tara Murphy, and James R. Curran. 2009. Named entity recognition in Wikipedia. In *Proceedings of the 2009 Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources (People's Web)*, pages 10–18, Suntec, Singapore. Association for Computational Linguistics.

Jonathan Baxter. 2000. A model of inductive bias learning. *Journal of Artificial Intelligence Research (JAIR)*, 12(1):149–198.

Shai Ben-David and Reba Schuller. 2003. Exploiting task relatedness for multiple task learning. In *Learning Theory and Kernel Machines*, pages 567–580, Berlin, Heidelberg. Springer Berlin Heidelberg.

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014. NoSta-d named entity annotation for German: Guidelines and dataset. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2524–2531, Reykjavik, Iceland. European Language Resources Association (ELRA).

Joachim Bingel and Anders Søgaard. 2017. Identifying beneficial task relations for multi-task learning in deep neural networks. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 164–169, Valencia, Spain. Association for Computational Linguistics.

Johannes Bjerva. 2017. Will my auxiliary tagging task help? estimating auxiliary tasks effectivity in multi-task learning. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 216–220, Gothenburg, Sweden. Association for Computational Linguistics.

Johannes Bjerva, Barbara Plank, and Johan Bos. 2016. Semantic tagging with deep residual networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3531–3541, Osaka, Japan. The COLING 2016 Organizing Committee.

BNC Consortium. 2007. The British National Corpus, version 3 (BNC XML Edition).

Johan Bos, Valerio Basile, Kilian Evang, Noortje Venhuizen, and Johannes Bjerva. 2017. The Groningen Meaning Bank. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, volume 2, pages 463–496. Springer.

Rich Caruana. 1997. Multitask learning. *Machine Learning*, 28(1):41–75.

Soravit Changpinyo, Hexiang Hu, and Fei Sha. 2018. Multi-task learning for sequence tagging: An empirical study. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2965–2977, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Thomas M. Cover and Joy A. Thomas. 2006. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, New York, New York, USA.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ábel Elekes, Martin Schäler, and Klemens Böhm. 2017. On the various semantics of similarity in word embedding models. In *Proceedings of the 17th ACM/IEEE Joint Conference on Digital Libraries*, JCDL '17, pages 139–148, Toronto, Ontario, Canada. IEEE Press.

Manaal Faruqui and Sebastian Padó. 2010. Training and evaluating a German named entity recognizer with semantic generalization. In *Semantic Approaches in Natural Language Processing: Proceedings of the 10th Conference on Natural Language Processing, KONVENS 2010*, pages 129–133, Saarbrücken, Germany.

Ivan Habernal and Iryna Gurevych. 2017. Argumentation mining in user-generated web discourse. *Computational Linguistics*, 43(1):125–179.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Suzana Ilić, Edison Marrese-Taylor, Jorge Balazs, and Yutaka Matsuo. 2018. Deep contextualized word representations for detecting sarcasm and irony. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 2–7, Brussels, Belgium. Association for Computational Linguistics.

Lukasz Kaiser, Aidan N. Gomez, Noam Shazeer, Ashish Vaswani, Niki Parmar, Llion Jones, and Jakob Uszkoreit. 2017. One model to learn them all. *arXiv:1706.05137*.

Joo-Kyung Kim, Young-Bum Kim, Ruhi Sarikaya, and Eric Fosler-Lussier. 2017. Cross-lingual transfer learning for POS tagging without cross-lingual resources. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2832–2838, Copenhagen, Denmark. Association for Computational Linguistics.

Young-Bum Kim, Karl Stratos, Ruhi Sarikaya, and Minwoo Jeong. 2015. New transfer learning techniques for disparate label sets. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 473–482, Beijing, China. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proceedings*, San Diego, California, USA.

T. O. Kvalseth. 1987. Entropy and correlation: Some comments. *IEEE Transactions on Systems, Man, and Cybernetics*, 17(3):517–519.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, California, USA. Morgan Kaufmann Publishers Inc.

Ran Levy, Ben Bogin, Shai Gretz, Ranit Aharonov, and Noam Slonim. 2018. Towards an argumentative content search engine using weak supervision. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2066–2081, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Liyuan Liu, Jingbo Shang, Frank F. Xu, Xiang Ren, Huan Gui, Jian Peng, and Jiawei Han. 2018. Empower Sequence Labeling with Task-Aware Neural Language Model. In *Proceedings of the Thirty-Second Conference on Artificial Intelligence (AAAI-2018)*, pages 5253–5260, New Orleans, Louisiana, USA.

Zhenqiu Liu, Zhongmin Guo, and Ming Tan. 2008. Constructing tumor progression pathways and biomarker discovery with fuzzy kernel kmeans and dna methylation data. *Cancer informatics*, 6:1–7.

Mitchell P. Marcus, Beatrice Santorini, Mary Ann Marcinkiewicz, and Ann Taylor. 1999. Penn Treebank 3 LDC99T42. Web Download. Philadelphia: Linguistic Data Consortium.

Héctor Martínez Alonso and Barbara Plank. 2017. When is multitask learning effective? semantic sequence prediction under varying data conditions. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 44–53, Valencia, Spain. Association for Computational Linguistics.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97, Sofia, Bulgaria. Association for Computational Linguistics.

Marina Meilă. 2005. Comparing clusterings: An axiomatic view. In *Proceedings of the 22Nd International Conference on Machine Learning*, ICML '05, pages 577–584, Bonn, Germany. ACM.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations (ICLR), Workshop Track Proceedings*, Scottsdale, Arizona, USA.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in pre-training distributed word representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

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 the 54th Annual Meeting of the Association for Computational Linguistics*

(Volume 2: Short Papers), pages 412–418, Berlin, Germany. Association for Computational Linguistics.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv:1706.05098*.

Sebastian Ruder, Joachim Bingel, Isabelle Augenstein, and Anders Søgaard. 2019. Latent multi-task architecture learning. In *Proceedings of the Thirty-Third Conference on Artificial Intelligence (AAAI-2019)*, pages 4822–4829, Honolulu, Hawaii, USA. Association for the Advancement of Artificial Intelligence.

Julio Cesar Salinas Alvarado, Karin Verspoor, and Timothy Baldwin. 2015. Domain adaption of named entity recognition to support credit risk assessment. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 84–90, Parramatta, Australia.

Claudia Schulz, Steffen Eger, Johannes Daxenberger, Tobias Kahse, and Iryna Gurevych. 2018. Multi-task learning for argumentation mining in low-resource settings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 35–41, New Orleans, Louisiana. Association for Computational Linguistics.

Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel Bowman, Miriam Connor, John Bauer, and Chris Manning. 2014. A gold standard dependency corpus for English. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2897–2904, Reykjavik, Iceland. European Language Resources Association (ELRA).

Anders Søgaard and Yoav Goldberg. 2016. Deep multi-task learning with low level tasks supervised at lower layers. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 231–235, Berlin, Germany. Association for Computational Linguistics.

Christian Stab and Iryna Gurevych. 2017. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659.

Alexander Strehl and Joydeep Ghosh. 2003. Cluster ensembles — a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research (JMLR)*, 3:583–617.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CoNLL '03, pages 142–147, Edmonton, Canada. Association for Computational Linguistics.

Nguyen Xuan Vinh, Julien Epps, and James Bailey. 2010. Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research (JMLR)*, 11:2837–2854.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, Mohammed El-Bachouti, Robert Belvin, and Ann Houston. 2013. OntoNotes Release 5.0 LDC2013T19. Web Download. Philadelphia: Linguistic Data Consortium.

Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. 2017. Transfer learning for sequence tagging with hierarchical recurrent networks. In *5th International Conference on Learning Representations, ICLR 2017, Conference Track Proceedings*, Toulon, France.

Yiyu Yao. 2003. *Information-Theoretic Measures for Knowledge Discovery and Data Mining*, pages 115–136. Springer Berlin Heidelberg, Berlin, Heidelberg.

A Appendices

A.1 Examples for casting label similarity as a clustering comparison problem

Let the dataset D use simplified named entity recognition (NER) as task T and part-of-speech (POS) tagging as task T' having the label sets:

$$L = \{\texttt{ORG}\text{anization}, \texttt{PER}\text{son}, \texttt{LOC}\text{ation}, \texttt{OTH}\text{er}\}$$
$$L' = \{\texttt{NN}\ \text{noun}, \texttt{VB}\ \text{verb}, \texttt{DT}\ \text{determiner}, \texttt{X}\ \text{other}\}$$

Let dataset D contain the following two sentences:

ORG	ORG	ORG	OTH	OTH	OTH
NN	NN	NN	VB	DT	NN
Walt	Disney	Productions	created	the	cartoon

OTH	PER	PER
NN	NN	NN
character	Donald	Duck

LOC	OTH	OTH	OTH	OTH	OTH	LOC
NN	VB	DT	X	NN	X	NN
Berlin	is	a	large	city	in	Germany

Table 5 shows the contingency table filled with the counts from both example sentences. The last row resp. column shows the sum of the counts in each column resp. row. The count $c_{\text{ORG,NN}}$ is three because there are exactly three tokens (*Walt Disney Productions*) tagged both ORG and NN. Other label-pairs are derived analogously from the remaining tokens of the dataset D.

With Equation 6, the normalized mutual information can be calculated from the counts in the contingency table. Note that the logarithm is only defined for positive values, but the counts

	NN	VB	DT	X	Σ
ORG	3	0	0	0	3
PER	2	0	0	0	2
LOC	2	0	0	0	2
OTH	3	2	2	2	9
Σ	10	2	2	2	16

Table 5: Counts from example dataset D for comparison of NER and POS tagsets

c_{ij} are often zero. The convention $0 \log(0) = 0$ is used to mitigate this issue because $x \log(x) \to 0$ when $x \to 0$. The normalized mutual information for the data in Table 5 can now be calculated: $I(L; L') = 0.437893$ and $H(L, L') = 2.78064$. Finally, $NMI_{joint} = 0.157479$.

A.2 Examples for the text overlap approach

Below are two example datasets annotated with the reduced POS tagset introduced previously:

(10)
VB	DT	NN	X	VB	DT
Creating	an	example	to	explain	the
NN	VB	DT	X	NN	X	X
process	is	an	impossible	task	.	To
VB	DT	NN	X	NN	X	NN
process	the	data	,	counts	of	words
X	NN	VB	VB	X		
and	labels	are	needed	.		

(11)
X	VB	DT	NN	X	DT	X
This	is	the	data	for	the	second
NN	X	DT	NN	X	VB	DT	X
dataset	.	The	process	to	find	the	right
NN	X	X	NN	VB	DT	NN
words	for	this	example	took	a	second
X						
.						

Table 6 shows the two Datasets 10 and 11 after the transformation. In the examples, the words *process* and *second* are ambiguous without context and thus have multiple labels. Table 7 shows the result of the *additive* method to combine the label counts from both datasets. The word *example* occurs once in each dataset and is both times tagged as NN. In the contingency table the count for (NN, NN), i.e. row 2 column 2, is increased by two. The word *the* occurs two resp. three times in the datasets and is always labeled DT. Consequently, the count in the contingency table at (DT, DT), i.e. row 1 column 1, is increased by five. For *process*, an issue is that

Word	#	DT	NN	VB	X
example	1	0	1	0	0
to	1	0	0	0	1
the	2	2	0	0	0
process	2	0	1	1	0
is	1	0	0	1	0
.	2	0	0	0	2
data	1	0	1	0	0
words	1	0	1	0	0

(a) Counts for words and their labels in Dataset 10

Word	#	DT	NN	VB	X
is	1	0	0	1	0
the	3	3	0	0	0
data	1	0	1	0	0
.	2	0	0	0	2
process	1	0	1	0	0
to	1	0	0	0	1
words	1	0	1	0	0
example	1	0	1	0	0

(b) Counts for words and their labels in Dataset 11

Table 6: Transformation of word-label pairs to an associated count-based representation. Only words occurring in both datasets are shown.

it has multiple labels in the first dataset: NN and VB. In the second dataset, there is only a single occurrence of *process* with label NN. The counts in the contingency table are increased by two for the positions (NN, NN) and (VB, NN). However, the single occurrence is now used twice. An improvement is to split the counts by the number of labels in the other dataset, so that the two affected positions are not increased by two but by 1.5.

A.3 Examples for the vector space approach

Applying the extension using word embeddings on the two example Datasets 10 and 11 would use the words not occurring in both datasets. *Creating* from Dataset 10 might have the closest match with *process* from Dataset 11. Thus, the count for (VB, NN) would be increased, which clearly is a mismatch. The word *an* might have the lowest vector space distance to *a* from the other dataset. This accurate match would increase the count for (DT, DT). The remaining, so far unused, words from Dataset

	DT	NN	VB	X	Σ
DT	5	0	0	0	5
NN	0	8	0	0	8
VB	0	2	2	0	4
X	0	0	0	6	6
Σ	5	10	2	6	23

Table 7: Contingency table derived from the additive combination of counts in Table 6.

10 have to be matched with their most similar counterparts from Dataset 11. For each pair of words, the count for the corresponding label-pair needs to be increased in the contingency table. While most vector representation matches between those two example datasets are inadequate, the quality of these matches is higher with larger datasets.

The application of the token-based approach using contextual embeddings on the two example Datasets 10 and 11 would work in the following way. All tokens in the two datasets are augmented with their corresponding contextual vector representations, thereby creating an associative array from a numeric vector to a label. For each word embedding in the first dataset, the vector representation with the closest distance from the other dataset is selected. Assuming the five matches are *Creating–is*, *an–the*, *example–data*, *to–for* and *explain–is*, the counts in a contingency table have to be increased for the label-pairs (`VB`, `VB`), (`DT`, `DT`), (`NN`, `NN`), (`X`, `X`) and (`VB`, `VB`).

A.4 Additional scores for the controlled environment experiments

The shared vocabulary values shown in Figure 3 exhibit a clear diagonal line of maximal shared vocabulary due to pairs of identical dataset samples. The remaining values are in accordance with the dataset sizes. For a chosen dataset, the shared vocabulary ratio increases with the size of the second dataset used in the comparison. Thus, there is no systematic difference between POS tagging and NER datasets nor a clear distinction between samples within the same original dataset and other datasets. Overall, the shared vocabulary is unsuitable to select datasets deemed similar.

Figure 4 shows the joint label entropy obtained from the same contingency tables as the NMI

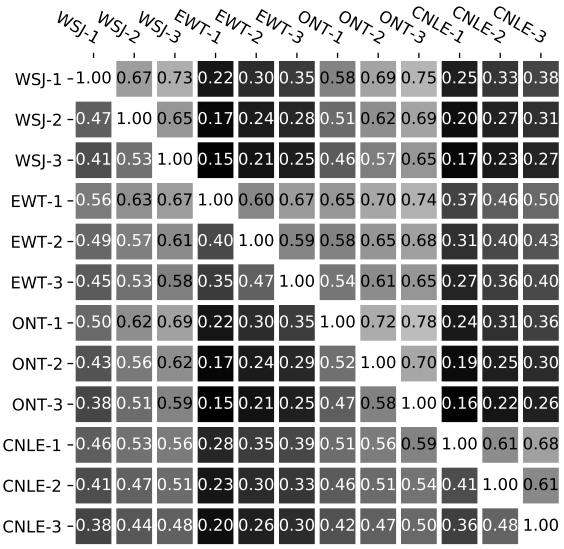

Figure 3: Pairwise shared vocabulary ratio (Equation 8) between the twelve sampled datasets. The heat map encodes the values from 0.0 in black to 1.0 in white.

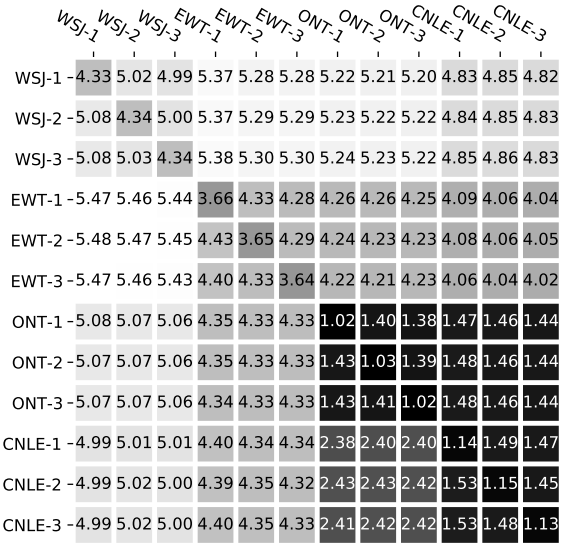

Figure 4: Pairwise joint label entropy values (Equation 2 and denominator of Equation 6) obtained on contingency tables filled with the vector space similarity approach using contextual BERT embeddings. The heat map encodes the values from min in black to max in white.

scores presented in Figure 2. While pairs of identical datasets exhibit a lower entropy relative to other pairs in the same row or column, there is no way to distinguish samples of the same original dataset from any other. The entropy values for NER–NER pairs are by far lower than any other pairs. This is reasonable as the "O" labels by far make up the majority of all labels in NER datasets. However, this does not help to find similar dataset in other cases, because there is no meaningful ordering of the entropy values when comparing any of the POS samples with all the other samples. In short, joint label entropy is not appropriate to find datasets deemed similar.

A.5 Neural network training procedure and hyperparameters

We train each model for at most 100 epochs with an early-stopping patience of 10 and a batch size of 256. The main and auxiliary training datasets are combined via interleaved batches from both datasets. Due to negligible effect, the dimensions of the character embeddings and hidden units are fixed at 32 resp. 64. 128 and 256 dimensions are tested for the word embeddings and the hidden units of the word GRU that can have either one or two layers. We use the Adam (Kingma and Ba, 2015) optimizer.

For POS tagging, the learning rate is fixed at 0.002. The best dropout value is chosen from the values 0, 0.25, 0.5, 0.75. Additional regularization via weight decay is selected from the values 0, 0.1, 0.01, 0.001.

For NER, the learning rate is set to 0.005 and weight decay uses a fixed value of 0.05. The range for dropout is narrowed to the values $0.3, 0.4, 0.5, 0.6$. Each combination of hyperparameters is run with three random seeds to mitigate performance fluctuations due to the random initialization of the network weights. While the POS tagging experiments only used a Softmax classifier, we evaluate both Softmax and CRF classifiers for NER.