Assessing the Difficulty of Classifying ConceptNet Relations in a Multi-Label Classification Setting

Maria Becker, Michael Staniek, Vivi Nastase, and Anette Frank
Heidelberg University, Department of Computational Linguistics
{mbecker/staniek/nastase/frank}@cl.uni-heidelberg.de

Abstract

Commonsense knowledge relations are crucial for advanced NLU tasks. We examine the learnability of such relations as represented in CONCEPTNET, taking into account their *specific properties*, which can make relation classification difficult: a given concept pair can be linked by multiple relation types, and relations can have multi-word arguments of diverse semantic types. We explore a neural *open world multi-label classification approach* that focuses on the evaluation of classification accuracy for individual relations. Based on an in-depth study of the specific properties of the CONCEPTNET resource, we investigate the impact of different relation representations and model variations. Our analysis reveals that the complexity of argument types and relation ambiguity are the most important challenges to address. We design a customized evaluation method to address the incompleteness of the resource that can be expanded in future work.

1 Introduction

Commonsense knowledge can be seen as a large amount of diverse but simple facts about the world, people and everyday life, e.g., *Cars are used to travel* or *Birds can fly* (Liebermann, 2008). Commonsense knowledge obtained from CONCEPTNET is increasingly used in advanced NLU tasks, such as textual entailment (Weissenborn et al., 2018), reading comprehension (Mihaylov and Frank, 2018), machine comprehension (Wang and Li, 2018; José-Angel González and Hurtado Oliver, Lluís and Segarra, Encarna and Pla, Ferran, 2018), question answering (Ostermann et al., 2018) or dialogue modeling (Young et al., 2018) and also applications in vision (Le et al., 2013). Some of these approaches exploit embeddings learned from CONCEPTNET, others select specific relations from it, depending on the application.

This paper proposes a multi-label neural approach for classifying CONCEPTNET relations, where the task is to predict one (or several) commonsense relations from a given set of relation types that hold between two given concepts from CONCEPTNET. In future work, the predicted relations can then be used for enriching CONCEPTNET by adding relations between concepts which are not yet linked in the network.

We design the task of multi-label neural relational classification to account for specific properties of CONCEPTNET:

(i) CONCEPTNET's relation inventory is not designed to be disjunct: a given pair of relation arguments (in CONCEPTNET: *concepts*) may be connected by more than one relation type: e.g. ⟨*people*,DESIRES/CAPABLEOF,*eating in groups*⟩, ⟨*reading*,USEDFOR/CAUSES,*education*⟩. This places relations in close vicinity in semantic space, making relation prediction a hard task.

(ii) Concepts often are *multi-word expressions* of *different phrase types* (e.g., noun or verb phrases), posing a challenge for argument representation. Relation slots may also be filled by *different semantic types*: e.g., the 2nd argument of DESIRES can be an entity or event. Such heterogeneous signatures increase classification difficulty.

(iii) As any knowledge resource, CONCEPTNET is incomplete, which means that relations between concepts are missing. The incompleteness of the resource poses serious evaluation problems, since assumed negative instances may in fact be positive.

To tackle these issues we perform a thorough experimental examination of the learnability of CONCEPTNET relations in a controlled multi-label classification setting. Our contributions are: (i) a cleaned and balanced data subset covering the 14 most frequent relation types from the core part of CONCEPTNET that serves as a basis for assessing relation-specific classification performance. We extend this dataset to an open-world classification setup; (ii) a neural multi-label classification approach with various model options for the representation of relations and their (multi-word) arguments, including relation-specific label prediction thresholds; (iii) an in-depth analysis of specific properties of the CONCEPTNET relation inventory, from which we derive hypotheses that we evaluate in classification experiments; (iv) we perform detailed analysis of results that confirm a great number of our hypotheses regarding specific classification challenges; (v) finally, we assess the amount of potential evaluation discrepancies due to the incompleteness of the resource in a small-scale annotation experiment.

2 Related Work

2.1 Semantic Relation Classification

Semantic relation classification covers a wide range of methods and learning paradigms for representing relation instances (see Nastase et al. 2013 for an overview). Typically, the data is presented to the learner as independent instances, with or without a sentential context. Relation classification models represent the meaning of the arguments (attributional features) and if context is available, also the relation (relational features).

Recently Deep Learning has strongly influenced semantic relation learning. Word embeddings can provide attributional features for a variety of learning frameworks (Attia et al., 2016; Vylomova et al., 2016), and the sentential context – in its entirety, or only the structured (through grammatical relations) or unstructured phrase expressing the relation – can be modeled through a variety of neural architectures – CNN (Tan et al., 2018; Ren et al., 2018) or RNN variations (Zhang et al., 2018).

2.2 CONCEPTNET Relation Classification

Speer et al. (2008) introduce AnalogySpace, a representation of concepts and relations in CONCEPTNET built by factorizing a matrix with concepts on one axis and their features or properties (according to CONCEPTNET) on the other. This low-dimensional representation allows for finding analogous facts, generalizations, new categories and justifications for classifications based on known properties. While this representation allows for recomputing the confidence of existing facts, the focus was not on classifying or trying to learn specific relations represented in the resource.

Li et al. (2016) apply *matrix factorization* to CONCEPTNET with the aim of resource extension and report 91% accuracy in a *binary* evaluation (i.e., verifying the correctness of an (unlabeled) link between concepts). Saito et al. (2018) expand this work by combining the knowledge base completion task (distinguishing true relation triples consisting of arbitrary phrases from false ones) with the task of knowledge generation (finding the second entity for a given first entity and a given relation). They enhance the link prediction model of Li et al. with a model that learns the two tasks – knowledge base completion and knowledge generation – jointly and outperform the completion accuracy results of Li et al. by up to 3pp.

Many NLU tasks rely on *specific relations* from CONCEPTNET (Le et al., 2013; Shudo et al., 2016). It is thus important to assess classification accuracy for individual relation types.

3 The Difficulty of CONCEPTNET Relation Classification

3.1 CONCEPTNET Dataset

The Open Mind Common Sense (OMCS) project (Speer et al., 2008) started the acquisition of common sense knowledge from contributions over the web, leading to CONCEPTNET, which now also includes expert-created resources (such as WordNet) and automatically extracted knowledge or knowledge obtained through games with a purpose (Speer et al., 2008). The current version, CONCEPTNET 5.6, comprises 37 relations, some of which are commonly used in other resources like WordNet (e.g. ISA, PARTOF) while most others are more specific to capturing commonsense information and as such are particular to CONCEPTNET (e.g. HASPREREQUISITE, MOTIVATEDBYGOAL). With very few exceptions (e.g., SYNONYM or ANTONYM), CONCEPTNET-relations are asymmetric. The English version consists of 1.9 million concepts and 1.1 million links to other databases, such as DBpedia. In our work we focus on the English OMCS subpart (CN-OMCS).

3.2 Task Definition

Given a pair of concepts $\langle c_i, c_j \rangle$, where c_i, c_j may be multi-word expressions, the task is to automatically predict one (or several, see §3.3.2 for the multi-label aspect of the task) commonsense relations r_t from a given set of CONCEPTNET relation types R_{CN} that hold between c_i and c_j. Relations are presented to the classifier without textual context, and thus a crucial aspect is using a representation that properly captures the semantics of the arguments.

3.3 Designing a Relation Classification System for CONCEPTNET

CONCEPTNET has very specific properties in terms of the relations included, the type of the arguments, coverage and completeness. A successful relation classification system should take these into account. Given the heterogeneity of sources of CONCEPTNET, we focus on its core part, in particular CN-OMCS-CLN, a subset selected from CN-OMCS that includes ca. 180K triples from 36 relation types, restricted to known vocabulary from the GoogleNews Corpus (see §4.1 for further details).

3.3.1 Representing the Inputs

Word embeddings have been shown to provide useful semantic representations, capturing lexical properties of words and relative positioning in semantic space (Mikolov et al., 2013b), which has been exploited for semantic relation classification (Vylomova et al., 2016; Attia et al., 2016).

Following this work, we represent a *pair of concepts* $\langle c_i, c_j \rangle$ whose relation we want to classify through their embeddings v_{c_i} and v_{c_j}. These argument representations can be combined by subtraction $(v_{\mathrm{c_i}} - v_{\mathrm{c_j}})$ (*DiffVec*; cf. wee , rol), addition $v_{\mathrm{c_i}} + v_{\mathrm{c_j}}$ (*AddVec*) or concatenation $[v_{\mathrm{c_i}}, v_{\mathrm{c_j}}]$ (*ConcatVec*, cf. bar).

One of the issues in using such representations for CONCEPTNET is the fact that most CONCEPTNET concepts are multi-word expressions (1.93 words on average, cf. Table 2). We experiment with two ways of producing a representation for a multi-word concept: (i) computing a *centroid vector*, as the normalized sum over the embedding vectors of all words in the expression (as the baseline); (ii) encoding the expression using an RNN, e.g. a (Bi)LSTM, which encodes sequences of various lengths into one fixed-length vector. We hypothesize that using an RNN yields better concept representations than centroid vectors.

3.3.2 Constructing a Multi-Label Classifier

An important characteristic of CONCEPTNET is that more than one relation can hold for a given pair of concepts. On average this applies to 5.37% of instances per relation (cf. 2). Consequently, we cast our classification task as a *multi-label classification problem.*

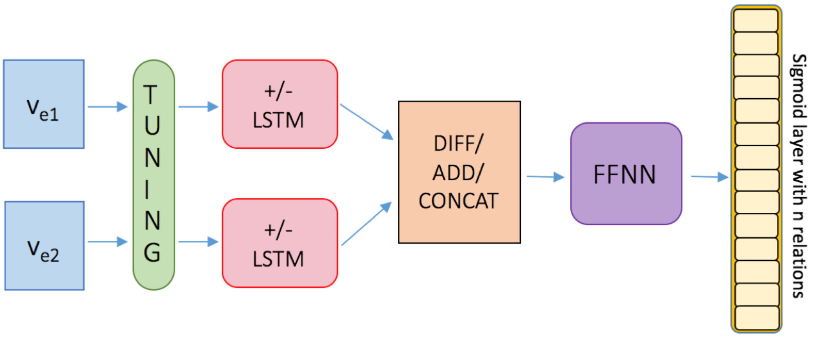

Figure 1: Multi-Label Classification Model.

Model architecture. Fig. 1 illustrates the model architecture. Input concept pairs are encoded – as centroids or using RNNs – and the representations are combined and presented to a feed-forward neural network (FFNN) with one hidden layer to non-linearly separate the relation classes.

In single-label classification, the probability for a class is not independent from the other class probabilities. Hence, $softmax$ is typically used at the output layer. By contrast, in multi-label classification, we want to model class predictions individually. The $sigmoid$ models the probability of a label as an independent *Bernoulli* distribution:

$$sig(t) = \frac{1}{1+e^{-t}} = \frac{1}{2}(1 + tanh\frac{t}{2})$$

This actually translates to an independent binary neural network for each label, resulting in a set of isolated binary classification tasks (cf. Sterbak 2017; He and Xia 2018).

The FFNN uses $sigmoid(\sigma(xW^h)W^o)$,where x is the input vector and W^h and W^o are weight matrices. We use binary cross entropy as our loss function. The architecture allows us to tune pre-trained embeddings for our relation learning task.

The hypotheses arising from the multi-label setting of CONCEPTNET are: (i) discrimination of overlapping classes is more difficult, compared to the usual relation classification task with disjoint relations (e.g. Hendrickx et al. 2010). (ii) given the incompleteness of CONCEPTNET, the classification performance may be erroneously assessed due to missing relations in the data. We will estimate the effect of this phenomenon in a small-scale annotation experiment.

3.3.3 Relation Classification Difficulty and Relation-specific Thresholding

CONCEPTNET relation types show great divergence with respect to their argument's semantic and phrase types, as shown in 2. About half of the relation arguments are nominal, entity-denoting concepts, with location as a specific entity type, half of them are event-type arguments. Several relations take different semantic types in a single argument position (e.g., HASSUBEVENT, CAUSES). Diversity of semantic types and phrase types – especially within a single argument position – is a challenge for relation learning. We expect classification to be more difficult on relations with a mixture of argument types. Because of this, different thresholds may be needed for predicting different relation types. We adopt a customized multi-label prediction setup where we tune thresholds separately for each relation type. We expect that individually tuned, relation-specific thresholds improve overall classification performance.

USEDFOR	33290	HASPREREQ.	16565	HASPROP.	5782	HAS1SUB.	2697
ATLOCATION	23874	ISA	15063	REC.ACTION	4494	DESIRES	2584
HASSUBEVENT	20518	CAUSES	12414	HASA	4118		
CAPABLEOF	18909	MOT.BYGOAL	7243	CAUSESDES.	3587		

Table 1: Number of instances per 14 most frequent relations in CN-OMCS-CLN.

	semantic type		phrase type		multiword	words per	multi-label
relations	ARG$_1$	ARG$_2$	ARG$_1$	ARG$_2$	concepts (%)	concept (avg)	relations (%)
IsA	entity	entity	NP	NP	46.01	1.78	1.17
HasA	entity	entity	NP	NP	56.36	1.96	1.07
AtLocation	entity	location	NP	NP	34.66	1.42	1.06
HasProperty	entity	property	NP	AP	41.41	1.80	1.31
UsedFor	entity	event	NP	VP	63.98	1.95	4.41
CapableOf	entity	event	NP	VP	58.21	1.89	1.17
ReceivesAction	entity	event	NP	VP	57.78	2.24	0.18
CausesDesire	entity	event	NP	VP	74.35	2.10	0.67
Desires	entity	ev/entity	VP	V/NP	40.02	1.68	2.79
Motiv.ByGoal	event	event	VP	VP	79.33	2.21	6.57
HasPrerequisite	event	event	VP	VP	80.27	2.23	9.86
HasFirstSubev.	event	ev/entity	VP	V/NP	83.89	2.27	36.26
HasSubevent	event	ev/entity	VP	V/NP	80.21	2.21	11.02
Causes	ev/entity	ev/entity	V/NP	V/NP	73.42	2.10	12.47
All relations					61.01	1.93	5.37

Table 2: 14 most frequent relations in Cn-Omcs-Cln: their semantic and phrase types (col. 2-5); percentage of multiword concepts (col. 6); average number of words per concept (col. 7); percentage of relation instances for which we find another relation instance in Cn-Omcs-Cln (col. 8).

4 Experiments

4.1 Dataset Construction

Cn-Omcs-cln Cn-Omcs contains noise in form of typos, unknown words, or words from other languages than English.[1] We check all relation triples in Cn-Omcs against the vocabulary of *word2vec* embeddings trained on part of the Google News dataset.[2] This embedding set contains vectors for 3 million words and phrases. We discard all relation triples from Cn-Omcs which contain words that do not appear in this set. The resulting dataset – Cn-Omcs-cln – contains 179.693 triples drawn from 36 different relations (relation distribution displayed in Table 1). The Other class comprises all relations from Cn-Omcs-cln with less than 2000 instances.

Cn-Omcs-14 Based on Cn-Omcs-Cln we construct our experimental dataset Cn-Omcs-14 – a balanced dataset still large enough for applying neural methods. We include all relations from Cn-Omcs-cln with more than 2000 instances, and downsample to the least frequent class – 2586 instances per relation. To select the "best" instances for testing and tuning, we sort the relation triples by their confidence score, as provided by ConceptNet. Inspired by Li et al. (2016) we select the 10% (258) most confident tuples per relation for testing, the next 10% for development, the remaining 80% (2068) for training, cf. Table 3.

Closed vs. Open World Setting. Learning to classify relations in a closed world setting is limited to the relation types present in the data. We want to design a system that is also able to detect whether *a relation exists* between concepts – but none of the provided ones, or whether *no relation* holds. We thus extend the data set with two classes: **Other** – containing concept pairs that *do stand* in a relation, yet not any of those present in the target relation set; and **Random** – containing concept pairs that are *not related*.

Instances for the Other class consist of a sample of triples from the 22 low-frequency relations that were not included in Cn-Omcs-14, these are the following relations: MadeOf, DBPedia, RelatedTo, Difference, LocatedNear, CreatedBy, NotUsedFor, FormOf, DerivedFrom, ObstructedBy, Synonym, PartOf, SymbolOf, NotDesires, HasContext, DefinedAs,

[1] We find a lot of Chinese words with the English tag *en* in cn-Omcs.

[2] About 100 billion words, cf. Mikolov et al. 2013a.

dataset	relations	train	dev	test
CN-OMCS-14 CW	14	28,952	3612	3612
CN-OMCS-14 OW-1	14+1	30,960	3870	3870
CN-OMCS-14 OW-2	14+2	33,088	4128	4128

Table 3: Dataset details and splits.

HasLastSubevent, ExternalURL, InstanceOf, NotCapableOf, and NotHasProperty.

Instances for the Random class are generated similarly to Vylomova et al. (2016): 50% of instances are *opposite pairs*, obtained by switching the order of concept pairs within the same relation; 50% are *corrupt pairs*, obtained by replacing one concept in a connected pair with a random concept from the same relation. Using *corrupt pairs* ensures that our model does not simply learn properties of the word classes, but instead is forced to encode relation instances. Random and Other are the same size as the individual target relations.

4.2 Experiment Setup

Experiments and Datasets. We experiment with two open world settings: in OW-1 we add only the Random class to Cn-Omcs-14, to investigate whether the classifier is able to differentiate related from non-related concept pairs. in OW-2 we add both Other and Random to Cn-Omcs-14, to investigate whether the classifier can also learn to predict that an unknown relation exists or that no relation holds. We also report results of the closed world setting where we exclude Other and Random. Each dataset is split into training (80%), dev (10%) and test (10%) (cf. Table 3).
Evaluation. We evaluate model performance in terms of F1 score for each relation. We report averaged weighted F1 scores over 5 runs.

4.3 Model Parameters

Embeddings. Based on preliminary experiments[3], we use 300-dim. skip-gram *word2vec* embeddings trained on part of the Google News dataset (100 billion words, Mikolov et al. 2013a). Embeddings are tuned during training.
Concept representation. Concept are encoded using centroid vectors or an RNN (cf. §3.3.1).
Relation representation. We use the *ConcatVec* representation (§3.3.1), which we determined to be the most useful in preliminary experiments.
Label prediction thresholds are tuned in two ways: (i) a global threshold for all relations and (ii) separately tuned thresholds for each relation.
Hyperparameter settings were determined on the devset. For encoding of multiword terms we use bi-LSTMs with one hidden layer and a cell size of 350 (perform better than GRUs and LSTMs). For the Ffnn we tune the hidden layer size and the activation function. Optimal hyperparameters are 200 (Ffnn), 100 (Ffnn+Rnn), and $ReLU$ for both Ffnn and Ffnn+Rnn.
Implementation. We implemented our models with *PyTorch* (Paszke et al., 2017).

4.4 Results

Table 4 summarizes the results in open (OW) and closed world (CW) settings.

The overall best performing model across all settings is Ffnn+Rnn (as opposed to Ffnn with centroid argument representations) with relation-specific label prediction thresholds (as opposed to one global threshold value). In the OW setting we achieve overall F1-scores of 0.68 (OW-1) and 0.65 (OW-2). The CW setting leads to best results with 0.71 F1. The models improve by 4pp (OW-1), 7pp (OW-2) and

[3] We additionally tested Numberbatch embeddings (Speer et al., 2017), GloVe embeddings trained on Wikipedia and Gigaword (Pennington et al., 2014), context2vec embeddings trained on UkWaC (Melamud et al., 2016). In our experiments we discovered that all of these alternatives perform worse than the *word2vec* embeddings.

Setting	OpenWorld OW-1		OpenWorld OW-2		Closed World	
Model	FF	FF+RNN	FF	FF+RNN	FF	FF+RNN
IsA	.58 (.57)	.62 (.60)	.51 (.51)	.60 (.57)	.64 (.63)	.67 (.67)
HasA	.67 (.66)	.80 (.79)	.53 (.52)	.79 (.77)	.73 (.72)	.80 (.78)
AtLocation	.69 (.68)	.78 (.78)	.63 (.61)	.74 (.72)	.77 (.75)	.84 (.83)
HasProperty	.66 (.65)	.81 (.80)	.62 (.61)	.78 (.77)	.67 (.67)	.84 (.83)
UsedFor	.76 (.75)	.78 (.77)	.79 (.78)	.76 (.76)	.78 (.78)	.79 (.78)
CapableOf	.61 (.61)	.67 (.65)	.56 (.56)	.65 (.64)	.61 (.60)	.71 (.71)
ReceivesAction	.82 (.82)	.91 (.91)	.77 (.77)	.90 (.90)	.87 (.86)	.93 (.93)
Caus.Des.	.87 (.87)	.90 (.88)	.86 (.85)	.87 (.87)	.87 (.86)	.92 (.90)
Desires	.91 (.85)	.94 (.92)	.73 (.65)	.93 (.88)	.87 (.83)	.88 (.94)
MoticatedByGoal	.61 (.60)	.56 (.55)	.56 (.55)	.59 (.59)	.60 (.59)	.64 (.61)
HasPrerequisite	.45 (.41)	.38 (.36)	.38 (.36)	.38 (.36)	.43 (.42)	.39 (.38)
HasFirstSubevent	.54 (.53)	.55 (.55)	.49 (.49)	.56 (.55)	.51 (.50)	.61 (.60)
HasSubevent	.24 (.22)	.26 (.16)	.17 (.15)	.24 (.15)	.21 (.21)	.24 (.20)
Causes	.60 (.59)	.57 (.56)	.59 (.58)	.61 (.60)	.61 (.60)	.61 (.61)
Other	-	-	.39 (.39)	.40 (.40)	-	-
Random	.61 (.58)	.59 (.54)	.62 (.61)	.53 (.49)	-	-
Weighted F1	.64 (.63)	.68 (.66)	.58 (.56)	.65 (.63)	.65 (.64)	.71 (.69)

Table 4: Weighted F1 results on Cn-Omcs-14. Main results obtained with relation-specific prediction thresholds (in brackets: results for global prediction threshold).

	multi word terms (%)	words/term (avg)	multi-label rel.(%)		multi word terms (%)	words/term (avg)	multi-label rel.(%)
IsA	43.43	1.72	1.70	HasA	55.37	1.88	0.46
AtLocation	36.02	1.43	1.86	HasProperty	40.21	1.75	0.62
UsedFor	64.38	1.90	3.25	CapableOf	55.91	1.83	2.86
ReceivesAction	55.91	2.21	0.15	CausesDesire	74.33	2.09	0.23
Desires	40.11	1.68	2.01	MotivatedByGoal	78.46	2.16	5.19
HasPrerequisite	77.88	2.15	13.70	HasFirstSubevent	84.78	2.30	10.84
HasSubevent	79.23	2.20	14.94	Causes	72.11	2.04	4.18
Other	53.49	1.82	9.52	Random	55.21	1.84	0

Table 5: Relation statistics on Cn-Omcs-14. Results for all relations (Ow-1): 60.01 % of MW terms, 1.94 (average number of words/term), and 4.77 % of relation instances with multiple labels.

6pp (CW) when replacing centroids with bi-LSTM encoded concept representations. Relation-specific thresholds improve results by 2pp (FFNN+RNN on OW-1, OW-2 and CW). Across all settings (CW, OW-1, OW-2) the best performing relations are: Desires (0.94), ReceivesAction (0.91), CausesDesire (0.90). We observe lowest F1-scores for HasSubevent (0.26, 0.24), HasPrerequisite (0.38, 0.39) and HasFirstSubevent (0.55, 0.61) in OW-1 and CW, respectively. The random and other classes have poor results overall. OW-2 with two OW classes performs worse than OW-1 with the single random class. The low results on the other class (0.40) could stem from its heterogeneity. The system finds it difficult to differentiate other and random.

5 Analysis

In this section we will discuss the hypotheses derived from our analysis of ConceptNet properties (§3.3), and based on that, to determine which approaches and representations are best suited for ConceptNet-based commonsense relation classification. To aid the discussion we produced Figures 2, 3, 4, and Table 5.

Fig. 2 plots differences in performance for each relation for the setting we wish to compare: *concept encoding* using centroids (FFnn) vs. RNNs (FFnn+Rnn) (blue), *global vs. relation-specific* prediction threshold (orange), and OW-1 vs. CW setting (grey).

Fig. 3 visualizes ambiguous – that means co-occurring – relations in our dataset in a symmetric heatmap. Fig. 4 displays interrelations between concept characteristics and model performance, based on our best performing system (FFNN+RNN+ind. tuned thresholds, OW-1). To observe correlations between clas-

sification performance and different measurable characteristics of the data in Fig. 4, we scaled the following values for each relation to a common range of 0 to 15: the percentage of multi-word terms (cf. Table 2) (grey), the average number of words per concept (cf. Table 2) (yellow), percentage of relation instances with multiple labels (cf. Table 2) (blue), best model performance on OW-1 (FFnn+Rnn with individually tuned thresholds, cf. Table 4) (red) and the corresponding relation-specific thresholds (green).
Table 5 gives relation statistics on Cn-Omcs-14 (as opposed to Table 2, which gives statistics for the complete version Cn-Omcs-Cln).

5.1 Representing Multi-word Concepts

We hypothesized that there is a correlation between the length of the arguments and model performance when encoding arguments with an RNN. We find no such correlation – the relations that benefit the most from using an RNN (Fig. 2: blue and Fig. 4: yellow, red) are not those with the longest arguments (cf. Table 5). Instead we find that the relations HasProperty, HasA, AtLocation, and ReceivesAction benefit most from concept encoding with a RNN, followed by CapableOf, IsA, Desires, CausesDesire and Has(First)Subevent with lower margins. The missing correlation can be confirmed by a very low Pearson's coefficient of only 0.05 between (1) improvements we get from enhancing FFnn with RNN (i.e., delta of F1 scores for FFnn vs. FFnn+Rnn; both with individually tuned thresholds) and (2) the average number of words per concepts (cf. Table 2).

5.2 Threshold Tuning & Model Performance

We hypothesized that relations would benefit from having individually tuned thresholds. Overall, the models with RNN encoding of concepts benefit more from threshold tuning than the basic FFnn. Regarding single relations (Fig. 2, orange bars), HasSubevent and the open world class Random benefit the most from individual threshold tuning (both with relatively low F1 scores). The individual thresholds vary considerably across relations (Fig. 4).

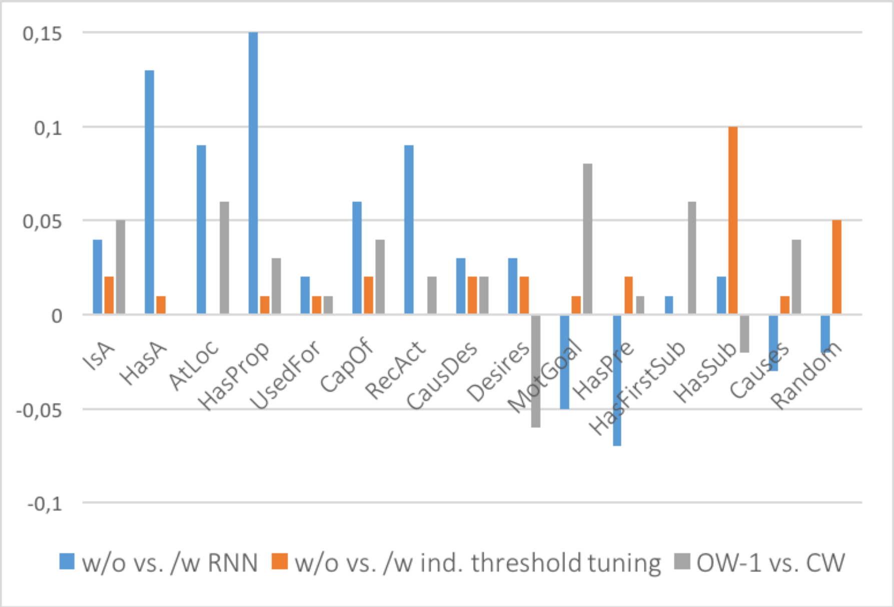

Figure 2: Delta of F1-scores on Cn-Omcs-14: (i) FFnn vs. FFnn+Rnn: (relation-specific threshold, OW-1, blue); (ii) global vs. relation-specific threshold (FFnn+Rnn,OW-1, orange); (iii) OW-1 vs. CW (FFnn+RNN, relation-specific threshold, grey).

To test whether relations that are harder to classify benefit the most from tuning the threshold (as the performance of HASSUBEVENT and RANDOM seem to indicate), we compute the correlation between (1) the difference of model performance with and without individually tuned thresholds (as described above) and (2) general model performance (F1 scores of FFNN+RNN with global threshold, OW-1). The score of -0.67 Pearson correlation indicates that indeed relations with lower general performance will tend to have higher improvements. This is also reflected in Fig. 4 (green and red), which also shows that for relations with higher F1 scores, higher thresholds tend to work better. Relation classification models applied to CONCEPTNET should therefore have higher thresholds for relations with high classification confidence (high F1 scores), while for relations with low performance lower thresholds are recommended.

5.3 Closed vs. Open World Setting

Most relations perform better in the CW setting (cf. grey bars in Fig. 2), especially MOTIVATEDBYGOAL, HASFIRSTSUBEVENT, ATLOCATION, and ISA (Fig. 2, grey). In contrast, DESIRES and HASSUBEVENT perform better in an open world setting (Fig. 2). Comparing the two settings OW-1 and OW-2 (Table 4, not displayed in Fig. 2), we find that only the relations MOTIVATEDBY, HASFIRSTSUBEVENT and CAUSES perform better in OW-2 than in OW-1. All other relations benefit from the OW-1 setting, especially ATLOCATION and the open world class RANDOM.

5.4 Relation Heterogeneity

We hypothesized that relations that are more heterogeneous with respect to the type of their arguments (whether semantic or phrasal) will be harder to learn. Comparing the degree of diversity of semantic or phrase types (Table 2) with model performance confirms this hypothesis. The relations that perform best have semantically or “phrasally” consistent arguments, whereas (apart from DESIRE) relation types that feature different types of entities or phrases in the same argument position tend to achieve low F1 scores.

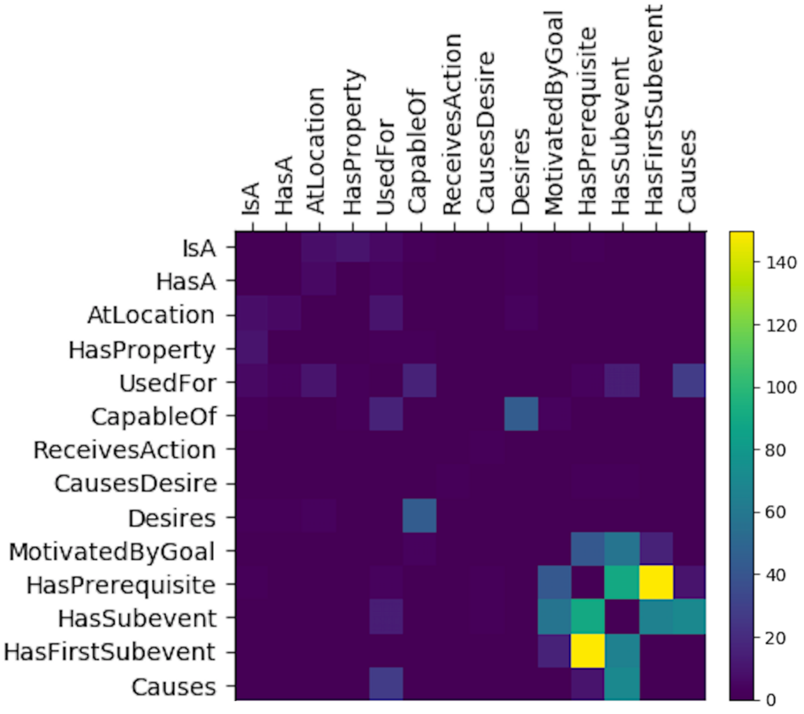

Figure 3: Visualizing ambiguous (co-occurring) relations in CN-OMCS-14 in a symmetric heatmap.

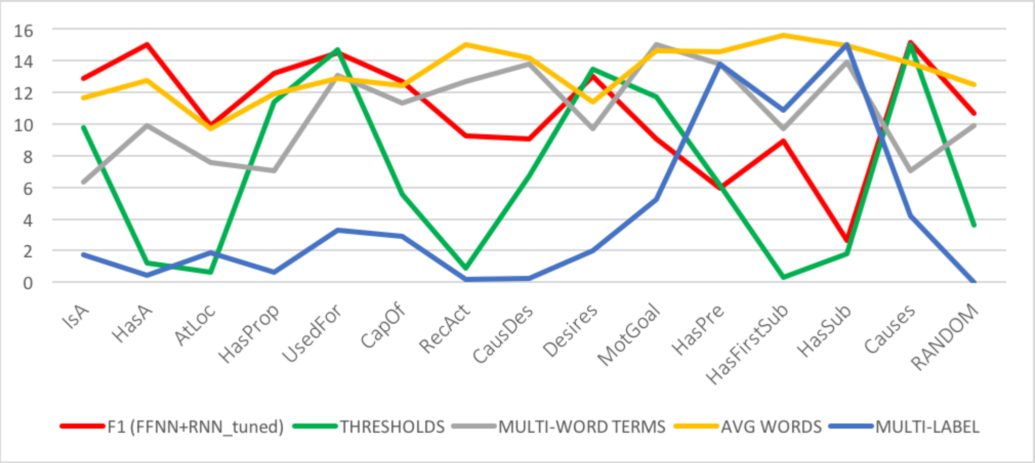

Figure 4: Interrelations between concept characteristics and model performance, based on FFNN+RNN+individually tuned thresholds, OW-1, scaled to range 0 to 15.

5.5 Relation Ambiguity

We hypothesized that relations that have multi-labeled instances (instances to which more than one label – relation – applies) will be more difficult to learn. Fig. 3 illustrates relation co-occurrences, i.e. relations that have overlapping instances. The most frequently co-occuring relations in CN-OMCS-14 are HASPREREQUISITE & HASFIRSTSUBEVENT, (150 co-occurrences), HASSUBEVENT & HASPREREQUISITE (90) and HASSUBEVENT & OTHER (86).[4] 399 concept pairs have two relation labels (e.g., ⟨*a cat,meow*⟩: DESIRES, CAPABLEOF), 20 pairs have three (e.g., ⟨*playing a harp,making music*⟩: CAUSES, HASSUBEVENT, USEDFOR), and two pairs have four: ⟨*opening a gift,surprise*⟩: HASPREREQUISITE, CAUSES, HASSUBEVENT, USEDFOR.

Fig. 4 shows a strong inverse correlation (-0.82 Pearson) between model performance and the number of multi-labeled instances for that relation.

5.6 Favorable vs. Unfavorable Properties of CONCEPTNET

We have investigated several variations of a relation classification model, each variation designed to mitigate some particular feature of CONCEPTNET relations. Analysis of these models have shown what impact each has on the model performance, and which issues we could address and which we could not. One of the issues was the length of the arguments. Using an RNN that can encode such sequences of various lengths did not lead to consistent improvements for relations with long arguments. The classifier still performs best on relations with short arguments. However, we do obtain overall better results with RNN encoding of arguments.

Another issue was the heterogeneity of relations in terms of the semantic or phrasal type of their arguments. The analysis has shown that indeed such relations suffer during classification, but individual tuning of the threshold partly helps.

One of the most striking challenges posed by the CONCEPTNET relation inventory remains the observed relation ambiguity. Here, our analysis matches our hypothesis, which was that multi-relation instances are harder to classify than relations for which we rarely find relation instances which co-occur with other relation labels. We further find that individual threshold tuning helps improving classification performance, especially for relations which are harder to classify and are characterized by low F1 scores. These are again exactly the relations which usually show other challenging properties including relation ambiguity, long arguments, and inner-relation diversity regarding concept and phrasal types.

[4] In CN-OMCS-CLN (complete, unbalanced dataset) the most frequently overlapping relations are: USEDFOR & CAUSES (800), HASSUBEVENT & CAUSES (636), and HASSUBEVENT & HASPREREQUISITE (628).

5.7 Impact of Missing Edges

The ambiguity of CONCEPTNET relations combined with the incompleteness of the resource pose challenges for evaluating the performance of a model. A classification decision marked as false positive could in fact be valid. This issue penalizes single-label and multi-label classifiers differently: a single-label classifier is not allowed to predict multiple labels, while a multi-label classifier will learn from potentially false negatives and depending on the distribution of the data could learn to over-predict. To investigate to what degree this issue impacts the results of our model, we manually annotate a small sample of the test data and compare it to the gold standard.

Annotation Experiment. We performed a small annotation experiment in which we manually control a subset of 200 instances from our test set for missing edges. Our sample consists of concept pairs which are related with one of the 14 relations in CN-OMCS-14, and we want to investigate if another, additional relation holds between the two concepts. We therefore present the concept pair and a randomly sampled relation from our relation set (excluding the gold label) to two annotators without showing the gold label. We ask them if the relation applies or not, and they are also allowed to assign *Not Sure* as a third option. The annotators agreed in 178 of 200 instances (91%). The annotations are merged by a third expert annotator. In the final gold version 18 (9%) of the instances are labelled as applicable (e.g. ⟨*cook dinner*,HASPREREQUISITE,*turn on stove*⟩), while 176 (88%) don't apply according to the annotators (e.g. ⟨*coffee*,HASSUBEVENT,*popular drink*⟩). According to this small annotated subset, we conclude that a lower bound of almost 9% of the predictions could be penalized due to incompleteness of the CONCEPTNET resource or our extracted subset, respectively. Of course this has to be verified in an annotation experiment of a larger scale.

6 Conclusion

In this paper we investigated several variations of a multi-label neural relational classifier for CONCEPTNET relations. Each variation was designed to account for specific properties of CONCEPTNET. An in-depth study revealed specific characteristics that can make CONCEPTNET relation classification difficult: several distinct relation types may hold for a given concept pair; some relations have heterogeneous arguments; and many concepts are expressed through multi-word terms. In light of these challenges posed by the specific properties of CONCEPTNET, we design a multi-label classification model which uses RNNs for representing multi-word arguments and individually tuned thresholds for improving model performance, especially for relations with unfavorable properties such as long arguments, relation ambiguity and inner-relation diversity. Our best performing model achieves F1 scores of 68 in an open world and 71 in a closed world setting. The analysis of the results in different configurations shows that the design decisions driven by multi-word representations and threshold tuning improved the overall classification performance, and that our model is able to tackle specific properties of CONCEPTNET. Yet, some challenges could not be resolved and need to be addressed in future work. In particular this concerns relation ambiguity and heterogeneity of relation arguments. The observed co-occurences of relations could be deployed for targeting relation ambiguity by building a meta classifier which learns which relations can or cannot occur together.

In future work, we plan to use the multi-label classification system proposed in this paper for enriching CONCEPTNET by predicting relations between concepts which are not yet linked in the network. Our investigation can further inform and caution the community on both the usefulness and the flaws of this resource and guide future work on using CONCEPTNET.

Acknowledgements. This work has been funded by Deutsche Forschungsgemeinschaft within the project *ExpLAIN. Between the Lines – Knowledge-based Analysis of Argumentation in a Formal Argumentation Inference System*, FR 1707/-4-1, as part of the RATIO Priority Program. We also want to thank our reviewers, our annotators Angel Daza and Esther van den Berg, and NVIDIA for donating GPUs.

References

Mohammed Attia, Suraj Maharjan, Younes Samih, Laura Kallmeyer, and Thamar Solorio. 2016. CogALex-V Shared Task: GHHH - Detecting Semantic Relations via Word Embeddings. In *Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex - V)*, pages 86–91.

Huihui He and Rui Xia. 2018. Joint Binary Neural Network for Multi-label Learning with Applications to Emotion Classification. *7th CCF International Conference. Hohhot, China.*, pages 250–259.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations between Pairs of Nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden. Association for Computational Linguistics.

José-Angel González and Hurtado Oliver, Lluís and Segarra, Encarna and Pla, Ferran. 2018. ELiRF-UPV at SemEval-2018 Task 11: Machine Comprehension using Commonsense Knowledge. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 1034–1037.

Dieu Thu Le, J Uijlings, and Raffaella Bernardi. 2013. Exploiting Language Models For Visual Recognition. *EMNLP 2013 - Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pages 769–779.

Xiang Li, Aynaz Taheri, Lifu Tu, and Kevin Gimpel. 2016. Commonsense Knowledge Base Completion. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1445–1455, Berlin, Germany. Association for Computational Linguistics.

Henry Liebermann. 2008. Usable AI Requires Commonsense Knowledge. In *Workshop on Usable artificial intelligence, held in conjunction with the Conference on Human Factors in Computing Systems (CHI)*.

Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. Context2Vec: Learning Generic Context Embedding with Bidirectional LSTM. In *CoNLL*, pages 51–61. Association for Computational Linguistics.

Todor Mihaylov and Anette Frank. 2018. Knowledgeable Reader: Enhancing Cloze-Style Reading Comprehension with External Common Knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, Melbourne, Australia. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Distributed Representations of Words and Phrases and Their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, pages 3111–3119, USA. Curran Associates Inc.

Tomas Mikolov, Scott Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic Regularities in Continuous Space Word Representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT-2013)*. Association for Computational Linguistics.

Vivi Nastase, Preslav Nakov, Diarmuid O Seaghdha, and Stan Szpakowicz. 2013. Semantic Relations between Nominals. *Synthesis lectures on human language technologies*, 6(1):1–119.

Simon Ostermann, Michael Roth, Ashutosh Modi, Stefan Thater, and Manfred Pinkal. 2018. SemEval-2018 Task 11: Machine Comprehension Using Commonsense Knowledge. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 747–757. Association for Computational Linguistics.

Adam Paszke, Sam Gross, and Adam Lerer. 2017. Automatic differentiation in PyTorch. In *Workshop Autodiff: The future of gradient-based machine learning software and techniques. Collocated with NIPS 2017.*

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, volume 14, pages 1532–1543.

Feiliang Ren, Di Zhou, Zhihui Liu, Yongcheng Li, Rongsheng Zhao, Yongkang Liu, and Xiaobo Liang. 2018. Neural Relation Classification with Text Descriptions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1167–1177. Association for Computational Linguistics.

Itsumi Saito, Kyosuke Nishida, Hisako Asano, and Junji Tomita. 2018. Commonsense Knowledge Base Completion and Generation. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 141–150. Association for Computational Linguistics.

Seiya Shudo, Rafal Rzepka, and Kenji Araki. 2016. Automatic Evaluation of Commonsense Knowledge for Refining Japanese ConceptNet. *Proceedings of the 12th Workshop on Asian Language Resources*, pages 105–112.

Robert Speer, Joshua Chin, and Catherine Havasi. 2017. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. *Proceedings of 31St AAAI Conference on Artificial Intelligence.*

Robert Speer, Catherine Havasi, and Henry Lieberman. 2008. AnalogySpace: Reducing the Dimensionality of Common Sense Knowledge. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 1*, pages 548–553. AAAI Press.

Tobias Sterbak. 2017. Guide To Multi-Class Multi-Label Classification With Neural Networks In Python. *Blogpost: https://www.depends-on-the-definition.com/guide-to-multi-label-classification-with-neural-networks/.*

Zhen Tan, Bo Li, Peixin Huang, Bin Ge, and Weidong Xiao. 2018. Neural Relation Classification Using Selective Attention and Symmetrical Directional Instances. *Symmetry*, 10:357.

Ekaterina Vylomova, Laura Rimell, Trevor Cohn, and Timothy Baldwin. 2016. Take and Took, Gaggle and Goose, Book and Read: Evaluating the Utility of Vector Differences for Lexical Relation Learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1671–1682, Berlin, Germany. Association for Computational Linguistics.

Zhigang Wang and Juanzi Li. 2018. Three-way Attention and Relational Knowledge for Commonsense Machine Comprehension. In *Proceedings of The 12th International Workshop on Semantic Evaluation*. Association for Computational Linguistics.

Dirk Weissenborn, Tomas Kocisky, and Chris Dyer. 2018. Dynamic Integration of Background Knowledge in Neural NLU Systems. *ICLR 2018.*

Tom Young, Erik Cambria, Iti Chaturvedi, Hao Zhou, Subham Biswas, and Minlie Huang. 2018. Augmenting End-to-End Dialog Systems with Commonsense Knowledge. *AAAI 2018*.

Xiaobin Zhang, Fucai Chen, and Ruiyang Huang. 2018. A Combination of RNN and CNN for Attention-based Relation Classification. In *Procedia Computer Science*, volume 131, pages 911 – 917.