A Report on the 2018 VUA Metaphor Detection Shared Task

Chee Wee Leong, Beata Beigman Klebanov
Educational Testing Service
{cleong,bbeigmanklebanov}@ets.org

Ekaterina Shutova
University of Amsterdam
e.shutova@uva.nl

Abstract

As the community working on computational approaches to figurative language is growing and as methods and data become increasingly diverse, it is important to create widely shared empirical knowledge of the level of system performance in a range of contexts, thus facilitating progress in this area. One way of creating such shared knowledge is through benchmarking multiple systems on a common dataset. We report on the shared task on metaphor identification on the VU Amsterdam Metaphor Corpus conducted at the NAACL 2018 Workshop on Figurative Language Processing.

1 Introduction

Metaphor use in everyday language is a way to relate our physical and familiar social experiences to a multitude of other subjects and contexts (Lakoff and Johnson, 2008); it is a fundamental way to structure our understanding of the world even without our conscious realization of its presence as we speak and write. It highlights the unknown using the known, explains the complex using the simple, and helps us to emphasize the relevant aspects of meaning resulting in effective communication. Consider the following examples of metaphor use in Table 1.

Metaphor has been studied in the context of political communication, marketing, mental health, teaching, assessment of English proficiency, among others (Beigman Klebanov et al., 2018; Gutierrez et al., 2017; Littlemore et al., 2013; Thibodeau and Boroditsky, 2011; Kaviani and Hamedi, 2011; Kathpalia and Carmel, 2011; Landau et al., 2009; Beigman Klebanov et al., 2008; Zaltman and Zaltman, 2008; Littlemore and Low, 2006; Cameron, 2003; Lakoff, 2010; Billow et al., 1997; Bosman, 1987); see chapter 7 in Veale et al. (2016) for a recent review.

M: *The alligator's teeth are like white* ***daggers***
I: The alligator have white and pointed teeth.

M: *He* ***swam*** *in a* ***sea*** *of diamonds.*
I: He is a rich person.

M: *Authority is a* ***chair****, it needs* ***legs*** *to* ***stand****.*
I: Authority is useless when it lacks support.

M: *In Washington, people change* ***dance partners*** *frequently, but not the* ***dance****.*
I: In Washington, people work with one another opportunistically.

M: *Robert Muller is like a* ***bulldog*** *— he will get what he wants.*
I: Robert Muller will work in a determined and aggressive manner to get what he wants.

Table 1: Metaphorical sentences (**M**) characterized by metaphors in bold and their literal interpretations (**I**)

In this paper, we report on the first shared task on automatic metaphor detection. By making available an easily accessible common dataset and framework for evaluation, we hope to contribute to the consolidation and strengthening of the growing community of researchers working on computational approaches to figurative language. By engaging a variety of teams to test their systems within a common evaluation framework and share their findings about more or less effective architectures, features, and data sources, we hope to create a shared understanding of the current state of the art, laying a foundation for further work.

This report provides a description of the shared task, dataset and metrics, a brief description of each of the participating systems, a comparative evaluation of the systems, and our observations about trends in designs and performance of the

Proceedings of the Workshop on Figurative Language Processing, pages 56–66
New Orleans, Louisiana, June 6, 2018. ©2018 Association for Computational Linguistics

systems that participated in the shared task.

2 Related Work

Over the last decade, automated detection of metaphor has become an increasingly popular topic, which manifests itself in both a variety of approaches and in an increasing variety of data to which the methods are applied. In terms of methods, approaches based on feature-engineering in a supervised machine learning paradigm explored features based on concreteness and imageability, semantic classification using WordNet, FrameNet, VerbNet, SUMO ontology, property norms, and distributional semantic models, syntactic dependency patterns, sensorial and vision-based features (Bulat et al., 2017; Köper and im Walde, 2017; Gutierrez et al., 2016; Shutova et al., 2016; Beigman Klebanov et al., 2016; Tekiroglu et al., 2015; Tsvetkov et al., 2014; Beigman Klebanov et al., 2014; Dunn, 2013; Neuman et al., 2013; Mohler et al., 2013; Hovy et al., 2013; Tsvetkov et al., 2013; Turney et al., 2011; Shutova et al., 2010; Gedigian et al., 2006); see Shutova et al. (2017) and Veale et al. (2016) for reviews of supervised as well as semi-supervised and unsupervised approaches.

Recently, deep learning methods have been explored for token-level metaphor detection (Rei et al., 2017; Gutierrez et al., 2017; Do Dinh and Gurevych, 2016). As discussed later in the paper later, the fact that all but one of the participating teams for the shared task experimented with neural network architectures testifies to the increasing popularity of this modeling approach.

In terms of data used for evaluating metaphor detection systems, researchers used specially constructed or selected sets, such as adjective noun pairs (Gutierrez et al., 2016; Tsvetkov et al., 2014), WordNet synsets and glosses (Mohammad et al., 2016), annotated lexical items (from a range of word classes) in sentences sampled from corpora (Özbal et al., 2016; Jang et al., 2015; Hovy et al., 2013; Birke and Sarkar, 2006), all the way to annotation of all words in running text for metaphoricity (Beigman Klebanov et al., 2018; Steen et al., 2010); Veale et al. (2016) review additional annotated datasets. By far the largest annotated dataset is the VU Amsterdam Metaphor Corpus; it has also been used for evaluating many of the cited supervised learning-based systems. Due to its size, availability, reliability of annotation, and popularity in current research, we decided to use it to benchmark the current field of supervised metaphor detection approaches.

3 Task Description

The goal of this shared task is to detect, at the word level, all metaphors in a given text. Specifically, there are two tracks, namely, **All Part-Of-Speech (POS)** and **Verbs**. The former track is concerned with the detection of all content words, i.e., nouns, verbs, adverbs and adjectives that are labeled as metaphorical while the latter track is concerned only with verbs that are metaphorical. We excluded all forms or *be*, *do*, and *have* for both tracks. Each participating individual or team can elect to compete in the All POS track, Verbs track, or both. The competition is organized into two phases: training and testing.

3.1 Dataset

We use the VU Amsterdam Metaphor Corpus (VUA) (Steen et al., 2010) as the dataset for our shared task. The dataset consists of 117 fragments sampled across four genres from the British National Corpus: **Academic**, **News**, **Conversation**, and **Fiction**. Each genre is represented by approximately the same number of tokens, although the number of texts differs greatly, where the news archive has the largest number of texts. We randomly sampled 23% of the texts from each genre to set aside for testing, while retaining the rest for training. The data is annotated using the MIP-VU procedure with a strong inter-annotator reliability of $\kappa > 0.8$. It is based on the MIP procedure (Group, 2007), extending it to handle metaphoricity through reference (such as marking *did* as a metaphor in *As the weather broke up, so did their friendship*) and allow for explicit coding of difficult cases where a group of annotators could not arrive at a consensus. The tagset is rich and is organized hierarchically, detecting various types of metaphors, words that flag the presense of metaphors, etc. In this paper, we consider only the top-level partition, labeling all content words with the tag "function=mrw" (metaphor-related word) as metaphors, while all other content words are labeled as non-metaphors. Table 2 shows the overall statistics of our training and testing sets.

To facilitate the use of the datasets and evaluation scripts beyond this shared task in future re-

Data	Training			Testing		
	#texts	#tokens	%M	#texts	#tokens	%M
Verbs						
Academic	12	4,903	31%	4	1,259	51%
Conversation	18	4,181	15%	6	2,001	15%
Fiction	11	4,647	25%	3	1,385	20%
News	49	3,509	42%	14	1,228	46%
All POS						
Academic	12	27,669	14%	4	6,076	24%
Conversation	18	11,994	10%	6	5,302	10%
Fiction	11	15,892	16%	3	4,810	14%
News	49	17,056	20%	14	6,008	22%

Table 2: Verbs and All POS datasets. The table reports the number of text fragments from BNC, number of tokens and percentage of tokens marked as metaphor group by genres.

search, the complete set of task instructions and scripts are published on Github[1]. Specifically, we provide a script to parse the original VUAMC.xml, which was not provided in our download bundle due to licensing restriction, to extract the verbs and other content words required for the shared task. We also provide a set of features used to construct the baseline classification model for prediction of metaphor/non-metaphor classes at the word level, and instructions on how to replicate the baselines.

3.2 Training phase

In this first phase, data is released for training and/or development of metaphor detection models. Participants can elect to perform cross-validation on the training data, or partition the training data further to have a held-out set for preliminary evaluations, and/or set apart a subset of the data for development/tuning of hyper-parameters. However the training data is used, the goal is to have N final systems (or versions of a system) ready for evaluation when the test data is released.

3.3 Testing phase

In this phase, instances for evaluation are released.[2] Each participating system generated predictions for the test instances, for up to N models.[3] Predictions are submitted to CodaLab[4] and evaluated automatically against the true labels. We selected CodaLab as a platform for organizing the task due to its ease of use, availability of communication tools such as mass-emailing, online forum for clarification of task issues, and tracking of submissions in real time. Submissions were anonymized. Hence, the only statistics displayed were the highest score of all systems per day, and the total number of system submissions per day. The metrics used for evaluation is the F1 score (least frequent class/label, which is "metaphor") with Precision and Recall also available via the detailed results link in CodaLab.

4 Systems

The shared task started on January 12, 2018 when the training data was made available to registered participants. On February 12, 2018, the testing data was released. Submissions were accepted until March 8, 2018. Overall, there were a total of 32 submissions by 8 unique individuals/teams for the Verbs track, and 100 submissions by 11 individuals/teams for the All POS track. All participants in the Verbs track also participated in the All POS track. In total, 8 system papers were submitted describing the algorithms and methodology for generating their metaphor predictions. In the following sections, we first describe the baseline classification models and their feature sets. Next, we report performance results and ranking of the best systems for each of the 8 teams. We also briefly describe the best-performing system for every team. The interested readers can refer to the

[1] https://github.com/EducationalTestingService/metaphor/tree/master/NAACL-FLP-shared-task

[2] In principle, participants could have access to the test data by independently obtaining the VUA corpus. The shared task was based on a presumption of fair play by participants.

[3] We set N=12.

[4] https://competitions.codalab.org/competitions/17805

teams' papers for more details.

Baseline Classifiers

We make available to shared task participants a number of features from prior published work on metaphor detection, including unigram features, features based on WordNet, VerbNet, and those derived from a distributional semantic model, POS-based, concreteness and difference in concreteness, as well as topic models.

As baselines, we train two logistic regression classifiers for each track (Verbs and All-POS), with instance weights inversely proportional to class frequencies. Lemmatized unigrams (UL) is a simple yet fairly strong baseline (**Baseline 1**). This feature is produced using NLTK (Bird and Loper, 2004) to generate the lemma of each word according to its tagged POS. As **Baseline 2**, we use the best system from Beigman Klebanov et al. (2016). The features are: lemmatized unigrams, generalized WordNet semantic classes, and difference in concreteness ratings between verbs/adjectives and nouns (UL + WordNet + CCDB).[5]

4.1 System Descriptions

The best-performing system from each participant is described below, in alphabetic order.

bot.zen (Stemle and Onysko, 2018) used word embeddings from different standard corpora representing different levels of language mastery, encoding each word in a sentence into multiple vector-based embeddings which are then fed into an LSTM RNN network architecture. Specifically, the backpropagation step was performed using weightings computed based on the logarithmic function of the inverse of the count of the metaphors and non-metaphors. Their implementation is hosted on Github[6] under the Apache License Version 2.0.

DeepReader (Swarnkar and Singh, 2018) The authors present a neural network architecture that concatenates hidden states of forward and backward LSTMs, with feature selection and classification. The authors also show that re-weighting examples and adding linguistic features (WordNet, POS, concreteness) helps improve performance further.

MAP (Pramanick et al., 2018) used a hybrid architecture of Bi-directional LSTM and Conditional Random Fields (CRF) for metaphor detection, relying on features such as token, lemma and POS, and using word2vec embeddings trained on English Wikipedia. Specifically, the authors considered contextual information within a sentence for generating predictions.

nsu_ai (Mosolova et al., 2018) used linguistic features based on unigrams, lemmas, POS tags, topical LDAs, concreteness, WordNet, VerbNet and verb clusters and trained a Conditional Random Field (CRF) model with gradient descent using the L-BFGS method to generate predictions.

OCOTA (Bizzoni and Ghanimifard, 2018) experimented with a deep neural network composed of a Bi-LSTM preceded and followed by fully connected layers, as well as a simpler model that has a sequence of fully connected neural networks. The authors also experiment with word embeddings trained on various data, with explicit features based on concreteness, and with preprocessing that addresses variability in sentence length. The authors observe that a model that combines Bi-LSTM with the explicit features and sentence-length manipulation shows the best performance. The authors also show that an ensemble of the two types of neural models works even better, due to a substantial increase in recall over single models.

Samsung_RD_PL (Skurniak et al., 2018) explored the use of several orthogonal resources in a cascading manner to predict metaphoricity. For a given word in a sentence, they extracted three feature sets: concreteness score from the Brysbaert database, intermediate hidden vector representing the word in a neural translation framework, and generated logits of a CRF sequence tagging model trained using word embeddings and contextual information. Trained on the VUA data, the CRF model alone outperforms that of a GRU taking all three features.

THU NGN (Wu et al., 2018) created word embeddings using a pre-trained word2vec model and added features such as embedding clusterings and POS tags before using CNN and

[5] Baseline 2 is "all-16" in Beigman Klebanov et al. (2018).

[6] https://github.com/bot-zen/naacl_flp_st

Bi-LSTM to capture local and long-range dependencies for generating metaphorical labels. Specifically, they used an ensemble strategy in which iterative modeling is performed by training on randomly selected training data and averaging the model predictions for finalized outputs. At the inferencing layer, the authors showed that the best-performing system is one achieved by using a weighted-softmax classifier rather than the Conditional Random Field predictor, since it can significantly improve the recall.

ZIL IPIPAN (Mykowiecka et al., 2018) used word2vec embeddings over ortographical word forms (no lemmatization) as an input for LSTM network for generating predictions. They explored augmenting word embeddings by binarized vectors that reflect the General Inquirer dictionary category of a word and its POS. Experiments were also carried out with different parametrization of LSTM based on type of unit network, number of layers, size of dropout, number of epochs, etc., though vectors enriched with POS information did not result in any improvement.

5 Results

Tables 3 and 4 show the performance and the ranking of all the systems, including the baseline systems. For overall results on All-POS track, three out of the seven systems outperformed the stronger of the two baselines, with the best submitted system gaining 6 F1-score points over the best baseline (0.65 vs 0.59). We note that the best system outperformed the baseline through improved precision (by 10 points), while the recall remained the same, around 0.7.

For the Verbs track, four out of the five systems outperformed both baselines. The best system posted an improvement of 7 F1-score points over best baseline (0.67 vs 0.60), achieved by improvements of about the same magnitude in both recall and precision.

In the following section, we inspect the performance of the different systems more closely.

6 Discussion

6.1 Trends in system design

All the submitted systems but one are based on a neural network architecture. Out of the top three systems that outperform the baseline on All-POS, two introduce explicit linguistic features into the architecture along with the more standard word-embedding-based representations, while the third experiments with using a variety of corpora – including English-language-learner-produced corpora – to compute word embeddings.

6.2 Performance across genres

Tables 3 and 4 show the overall performance for the best submission per team, as well as the performance of these systems by genre. It is clear that the overall F1 scores of 0.62-0.65 for the top three systems do not make explicit the substantial variation in performance across genres. Thus, Academic is the easiest genre, with the best performance of 0.74, followed by News (0.66), with comparable scores for Fiction (0.57) and Conversation (0.55). In fact, this trend holds not only for the top systems but for all systems, including baselines, apart from the lowest-performing system that showed somewhat better results on News than on Academic. The same observations hold for the Verb data. The large discrepancies in performance across different genres underscore the need for wide genre coverage when evaluating metaphor detection systems, as the patterns of metaphor use are quite different across genres and present tasks of varying difficulty to machine learning systems across the board.

Furthermore, we note that the best overall system, which is the only system that improves upon the baseline for every single genre in All-POS evaluation, improved over the baseline much more substantially in the lower-performance genres. Thus, for Academic and News, the increase is 1.4 and 5.2 F1 points, respectively, while the improvements for Conversation and Fiction are 8.1 and 11.1 points, respectively. The best-performing system thus exhibits more stable performance across genres than the baseline, though genre discrepancies are still substantial, as described above.

6.3 Part of Speech

6.3.1 AllPOS vs Verbs

We observe that for the four teams who improved upon the baseline on the Verbs-only track, their best performance on the Verbs was better than on the All-POS track, by 2.1-5 F1 score points.

Rank	Team	P	R	F1	Approach
All POS (Overall)					
1	THU NGN	0.608	0.700	0.651	word embeddings + CNN + Bi-LSTM
2	OCOTA	0.595	0.680	0.635	word embeddings + Bi-LSTM + linguistic
3	bot.zen	0.553	0.698	0.617	word embeddings + LSTM RNN
4	Baseline 2	0.510	0.696	0.589	UL + WordNet + CCDB + Logistic Regression
5	ZIL IPIPAN	0.555	0.615	0.583	dictionary-based vectors + LSTM
6	Baseline 1	0.521	0.657	0.581	UL + Logistic Regression
7	DeepReader	0.511	0.644	0.570	word embeddings + Di-LSTM + linguistic
8	Samsung_RD_PL	0.547	0.575	0.561	word embeddings + CRF + context
9	MAP	0.645	0.459	0.536	word embeddings + Bi-LSTM + CRF
10	nsu_ai	0.183	0.111	0.138	linguistic + CRF
All POS (Academic)					
1	THU NGN	0.725	0.746	0.735	word embedding + CNN + Bi-LSTM
2	Baseline 2	0.711	0.731	0.721	UL + WordNet + CCDB + Logistic Regression
3	Baseline 1	0.728	0.701	0.715	UL + Logistic Regression
4	bot.zen	0.743	0.681	0.711	word embeddings + LSTM RNN
5	OCOTA	0.724	0.695	0.709	word embeddings + Bi-LSTM + linguistic
6	ZIL IPIPAN	0.722	0.674	0.697	dictionary-based vectors + LSTM
7	DeepReader	0.641	0.682	0.661	word embeddings + Di-LSTM + linguistic
8	Samsung_RD_PL	0.649	0.631	0.640	word embeddings + CRF + context
9	MAP	0.743	0.526	0.616	word embeddings + Bi-LSTM + CRF
10	nsu_ai	0.283	0.100	0.148	linguistic + CRF
All POS (Conversation)					
1	THU NGN	0.453	0.711	0.553	word embeddings + CNN + Bi-LSTM
2	OCOTA	0.478	0.607	0.534	word embeddings + Bi-LSTM + linguistic
3	bot.zen	0.469	0.563	0.511	word embeddings + LSTM RNN
4	DeepReader	0.403	0.608	0.485	word embeddings + Di-LSTM + linguistic
5	MAP	0.503	0.456	0.478	word embeddings + Bi-LSTM + CRF
6	Baseline 2	0.334	0.809	0.472	UL + WordNet + CCDB + Logistic Regression
7	Samsung_RD_PL	0.505	0.439	0.470	word embeddings + CRF + context
8	Baseline 1	0.335	0.767	0.466	UL + Logistic Regression
9	ZIL IPIPAN	0.336	0.625	0.437	dictionary-based vectors + LSTM
10	nsu_ai	0.099	0.107	0.102	linguistic + CRF
All POS (Fiction)					
1	THU NGN	0.483	0.692	0.569	word embeddings + CNN + Bi-LSTM
2	OCOTA	0.460	0.631	0.532	word embeddings + Bi-LSTM + linguistic
3	bot.zen	0.474	0.569	0.517	word embeddings + LSTM RNN
4	DeepReader	0.414	0.597	0.489	word embeddings + Di-LSTM + linguistic
5	MAP	0.526	0.445	0.482	word embeddings + Bi-LSTM + CRF
6	ZIL IPIPAN	0.415	0.545	0.471	dictionary-based vectors + LSTM
7	Samsung_RD_PL	0.413	0.531	0.464	word embeddings + CRF + context
8	Baseline 2	0.366	0.614	0.458	UL + WordNet + CCDB + Logistic Regression
9	Baseline 1	0.372	0.594	0.457	UL + Logistic Regression
10	nsu_ai	0.121	0.120	0.120	linguistic + CRF
All POS (News)					
1	OCOTA	0.606	0.718	0.658	word embeddings + Bi-LSTM + linguistic
2	THU NGN	0.664	0.647	0.655	word embedding + CNN + Bi-LSTM
3	bot.zen	0.608	0.694	0.648	word embeddings + LSTM RNN
4	ZIL IPIPAN	0.649	0.578	0.612	dictionary-based vectors + LSTM
5	Baseline 2	0.567	0.650	0.606	UL + WordNet + CCDB + Logistic Regression
6	Baseline 1	0.591	0.593	0.592	UL + Logistic Regression
7	DeepReader	0.566	0.592	0.579	word embeddings + Di-LSTM + linguistic
8	Samsung_RD_PL	0.571	0.587	0.579	word embeddings + CRF + context
9	MAP	0.681	0.400	0.504	word embeddings + Bi-LSTM + CRF
10	nsu_ai	0.255	0.126	0.169	linguistic + CRF

Table 3: Performance and ranking of the best system per team and baselines for the All-POS track, including split by genre.

Rank	Team	P	R	F1	Approach
			Verbs (Overall)		
1	THU NGN	0.600	0.763	0.672	word embeddings + CNN + Bi-LSTM
2	bot.zen	0.547	0.779	0.642	word embeddings + LSTM RNN
3	ZIL IPIPAN	0.571	0.676	0.619	dictionary-based vectors + LSTM
4	DeepReader	0.529	0.708	0.605	word embeddings + Di-LSTM + linguistic
5	Baseline 2	0.527	0.698	0.600	UL + WordNet + CCDB + Logistic Regression
6	MAP	0.675	0.517	0.586	word embeddings + Bi-LSTM + CRF
7	Baseline 1	0.510	0.654	0.573	UL + Logistic Regression
8	nsu_ai	0.301	0.207	0.246	linguistic + CRF
			Verbs (Academic)		
1	Baseline 2	0.707	0.836	0.766	UL + WordNet + CCDB + Logistic Regression
2	DeepReader	0.684	0.865	0.764	word embeddings + Di-LSTM + linguistic
3	ZIL IPIPAN	0.752	0.768	0.760	dictionary-based vectors + LSTM
4	THU NGN	0.746	0.763	0.755	word embedding + CNN + Bi-LSTM
5	MAP	0.672	0.842	0.748	word embeddings + Bi-LSTM + CRF
6	Baseline 1	0.686	0.808	0.742	UL + Logistic Regression
7	bot.zen	0.769	0.617	0.685	word embeddings + LSTM RNN
8	nsu_ai	0.499	0.908	0.644	linguistic + CRF
			Verbs (Conversation)		
1	THU NGN	0.408	0.656	0.503	word embeddings + CNN + Bi-LSTM
2	bot.zen	0.355	0.729	0.477	word embeddings + LSTM RNN
3	DeepReader	0.366	0.605	0.456	word embeddings + Di-LSTM + linguistic
4	Baseline 2	0.301	0.821	0.441	UL + WordNet + CCDB + Logistic Regression
5	MAP	0.482	0.405	0.440	word embeddings + Bi-LSTM + CRF
6	ZIL IPIPAN	0.333	0.636	0.437	dictionary-based vectors + LSTM
7	Baseline 1	0.294	0.794	0.429	UL + Logistic Regression
8	nsu_ai	0.163	0.271	0.203	linguistic + CRF
			Verbs (Fiction)		
1	THU NGN	0.455	0.784	0.576	word embeddings + CNN + Bi-LSTM
2	bot.zen	0.411	0.766	0.535	word embeddings + LSTM RNN
3	MAP	0.538	0.513	0.525	word embeddings + Bi-LSTM + CRF
4	DeepReader	0.419	0.670	0.515	word embeddings + Di-LSTM + linguistic
5	Baseline 2	0.407	0.667	0.506	UL + WordNet + CCDB + Logistic Regression
6	ZIL IPIPAN	0.414	0.604	0.491	dictionary-based vectors + LSTM
7	Baseline 1	0.390	0.608	0.475	UL + Logistic Regression
8	nsu_ai	0.218	0.190	0.204	linguistic + CRF
			Verbs (News)		
1	THU NGN	0.694	0.744	0.718	word embedding + CNN + Bi-LSTM
2	bot.zen	0.667	0.764	0.712	word embeddings + LSTM RNN
3	Baseline 2	0.677	0.689	0.683	UL + WordNet + CCDB + Logistic Regression
4	ZIL IPIPAN	0.709	0.644	0.675	dictionary-based vectors + LSTM
5	DeepReader	0.644	0.665	0.654	word embeddings + Di-LSTM + linguistic
6	Baseline 1	0.668	0.619	0.643	UL + Logistic Regression
7	MAP	0.746	0.488	0.590	word embeddings + Bi-LSTM + CRF
8	nsu_ai	0.477	0.256	0.333	linguistic + CRF

Table 4: Performance and ranking of the best system per team and baselines for the Verbs track, including split by genre.

Team	All-POS	Verbs	Adjectives	Nouns	Adverbs	Best to Worst
THU NGN	.651	.674 (1)	.651 (2)	.629 (3)	.588 (4)	.09
OCOTA	.635	.669 (1)	.625 (2)	.609 (3)	.569 (4)	.10
bot.zen	.617	.655 (1)	.582 (3)	.594 (2)	.539 (4)	.12
Baseline 2	.589	.616 (1)	.557 (3)	.564 (2)	.542 (4)	.07
ZIL IPIPAN	.583	.619 (1)	.571 (2)	.552 (3)	.484 (4)	.14
Baseline 1	.581	.594 (1)	.578 (2)	.564 (3)	.563 (4)	.03
DeepReader	.570	.605 (1)	.568 (2)	.537 (3)	.521 (4)	.08
SamSung_RD_PL	.561	.615 (1)	.540 (2)	.516 (3)	.498 (4)	.12
MAP	.536	.586 (1)	.527 (2)	.481 (4)	.496 (3)	.10
nsu_ai	.138	.155 (1)	.131 (3)	.136 (2)	.102 (4)	.05
Av. rank among POS	–	1	2.3	2.8	3.9	.09
Rank order correlation with AllPOS performance	1	.94	.92	.98	.81	–

Table 5: Performance (F-score) of the best systems submitted to All-POS track by POS subsets of the test data. In parentheses, we show the rank of the given POS within all POS for the system. The last column shows the overall drop in performance from best POS (ranked 1) to worst (ranked 4).

This could be related to the larger preponderance of metaphors among verbs, which, in turn, leads to a more balanced class distribution in the Verbs data.

6.3.2 AllPOS by POS

To better understand performance patterns across various parts of speech, we break down the AllPOS test set by POS, and report performance of each of the best systems submitted to the AllPOS track on each POS-based subset of the test data; Table 5 shows the results. First, we observe that the average difference in performance between best and worst POS is 9 points (see column Best to Worst in the Table), with different systems ranging from 3 to 14. We note that the baseline systems are relatively more robust in this respect (3-7 points), while the the top 3 systems exhibit a 9-12 point range of variation in performance by POS. While this gap is substantial, it is much smaller than the 20-point gap observed in by-genre breakdown.

Second, we note that without exception all systems performed best on verbs, and for all but one system performance was worst on adverbs (see "Av. rank among POS" row in Table 5). Performance on adjectives and nouns was comparable for most systems, with slightly better results for adjectives for 7 out of 10 systems. These trends closely follow the proportions of metaphors within each POS:

While 30% of verbs are marked as metaphorical, only 8% of adverbs are thus marked, with nouns and adjectives occupying the middle ground with 13% and 18% metaphors, respectively.

Third, we observe that the relative performance of the systems is quite consistent across POS. Thus, the rank order correlation between systems' overall performance (AllPOS) and their performance on Verbs is 0.94; it is 0.98 for nouns and 0.92 for Adjectives (see the last row of Table 5). In fact, the top three ranks are occupied by the same systems in AllPOS, Verbs, Adjectives, and Nouns categories. The somewhat lower rank order correlation for Adverbs (0.81) reflects Baseline 1 (which ranks 6th overall) posting a relatively strong performance for Adverbs (ranks 3rd), while the ZIL IPIPAN system (ranks 5th overall) shows relatively weak performance on Adverbs (ranks 9th). Overall, the systems' relative standings are not much affected when parceled out by POS-based subsets.

7 Conclusion

This paper summarized the results of the 2018 shared task on metaphor identification in the VUA corpus, held as part of the 2018 NAACL Workshop on Figurative Language Processing. We provided brief descriptions of the participating systems for which detailed papers were submitted; systems' performance in terms of precision, recall, and F-score; and breakdowns of systems' performance by POS and genre.

We observed that the task of metaphor detection seems to be somewhat easier for verbs than for other parts of speech, consistently across participating systems. For genres, we observed a large discrepancy in best and worst performance, with results in the .7s for Academic and in .5s for Conversation data. Clearly, understanding and bridging the genre-based gap in performance is an important avenue for future work.

While most systems employed a deep learning architecture effectively, the baselines that use a traditional feature-engineering design were not far behind, in terms of performance; the stronger baseline came 4th overall. Indeed, some of the contributions explored a combination of a DNN architecture and explicit linguistic features; this seems like a promising direction for future work. Some of the teams made their implementations publicly available, which should facilitate further work on improving performance on this task.

8 Acknowledgements

As organizers of the shared task, we would like to thank all the teams for their interest and participation. Specifically, we would also like to thank Yuri Bizzoni for his help with pre-testing the shared task setup.

References

Beata Beigman Klebanov, Daniel Diermeier, and Eyal Beigman. 2008. Lexical cohesion analysis of political speech. *Political Analysis*, 16(4):447–463.

Beata Beigman Klebanov, Chee Wee Leong, and Michael Flor. 2018. A corpus of non-native written english annotated for metaphor. In *Proceedings of the Annual Meeting of the North American Chapter of the Association for Computational Linguistics*, New Orleans,LA.

Beata Beigman Klebanov, Chee Wee Leong, E Dario Gutierrez, Ekaterina Shutova, and Michael Flor. 2016. Semantic classifications for detection of verb metaphors. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 101–106.

Beata Beigman Klebanov, Chee Wee Leong, Michael Heilman, and Michael Flor. 2014. Different texts, same metaphors: Unigrams and beyond. In *Proceedings of the Second Workshop on Metaphor in NLP*, pages 11–17.

Richard M Billow, Jeffrey Rossman, Nona Lewis, Deberah Goldman, and Charles Raps. 1997. Observing expressive and deviant language in schizophrenia. *Metaphor and Symbol*, 12(3):205–216.

Steven Bird and Edward Loper. 2004. Nltk: the natural language toolkit. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, page 31. Association for Computational Linguistics.

Julia Birke and Anoop Sarkar. 2006. A clustering approach for nearly unsupervised recognition of nonliteral language. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.

Yuri Bizzoni and Mehdi Ghanimifard. 2018. Bigrams and bilstms: Two neural networks for sequential metaphor detection. In *Proceedings of the Workshop on Figurative Language Processing*, New Orleans,LA.

Jan Bosman. 1987. Persuasive effects of political metaphors. *Metaphor and Symbol*, 2(2):97–113.

Luana Bulat, Stephen Clark, and Ekaterina Shutova. 2017. Modelling metaphor with attribute-based semantics. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, volume 2, pages 523–528.

Lynne Cameron. 2003. *Metaphor in educational discourse*. A&C Black.

Erik-Lân Do Dinh and Iryna Gurevych. 2016. Token-level metaphor detection using neural networks. In *Proceedings of the Fourth Workshop on Metaphor in NLP*, pages 28–33.

Jonathan Dunn. 2013. What metaphor identification systems can tell us about metaphor-in-language. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 1–10.

Matt Gedigian, John Bryant, Srini Narayanan, and Branimir Ciric. 2006. Catching metaphors. In *Proceedings of the Third Workshop on Scalable Natural Language Understanding*, pages 41–48. Association for Computational Linguistics.

Pragglejaz Group. 2007. Mip: A method for identifying metaphorically used words in discourse. *Metaphor and symbol*, 22(1):1–39.

E Dario Gutierrez, Guillermo Cecchi, Cheryl Corcoran, and Philip Corlett. 2017. Using automated metaphor identification to aid in detection and prediction of first-episode schizophrenia. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2923–2930.

E Dario Gutierrez, Ekaterina Shutova, Tyler Marghetis, and Benjamin Bergen. 2016. Literal and metaphorical senses in compositional distributional semantic models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 183–193.

Dirk Hovy, Shashank Shrivastava, Sujay Kumar Jauhar, Mrinmaya Sachan, Kartik Goyal, Huying Li, Whitney Sanders, and Eduard Hovy. 2013. Identifying metaphorical word use with tree kernels. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 52–57.

Hyeju Jang, Seungwhan Moon, Yohan Jo, and Carolyn Rose. 2015. Metaphor detection in discourse. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 384–392.

Sujata S Kathpalia and Heah Lee Hah Carmel. 2011. Metaphorical competence in esl student writing. *RELC Journal*, 42(3):273–290.

Hossein Kaviani and Robabeh Hamedi. 2011. A quantitative/qualitative study on metaphors used by persian depressed patients. *Archives of Psychiatry and Psychotherapy*, 4(5-13):110.

Maximilian Köper and Sabine Schulte im Walde. 2017. Improving verb metaphor detection by propagating abstractness to words, phrases and individual senses. In *Proceedings of the 1st Workshop on Sense, Concept and Entity Representations and their Applications*, pages 24–30.

George Lakoff. 2010. *Moral politics: How liberals and conservatives think*. University of Chicago Press.

George Lakoff and Mark Johnson. 2008. *Metaphors we live by*. University of Chicago press.

Mark J Landau, Daniel Sullivan, and Jeff Greenberg. 2009. Evidence that self-relevant motives and metaphoric framing interact to influence political and social attitudes. *Psychological Science*, 20(11):1421–1427.

Jeannette Littlemore, Tina Krennmayr, James Turner, and Sarah Turner. 2013. An investigation into metaphor use at different levels of second language writing. *Applied linguistics*, 35(2):117–144.

Jeannette Littlemore and Graham Low. 2006. Metaphoric competence, second language learning, and communicative language ability. *Applied linguistics*, 27(2):268–294.

Saif Mohammad, Ekaterina Shutova, and Peter Turney. 2016. Metaphor as a medium for emotion: An empirical study. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 23–33.

Michael Mohler, David Bracewell, Marc Tomlinson, and David Hinote. 2013. Semantic signatures for example-based linguistic metaphor detection. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 27–35.

Anna Mosolova, Ivan Bondarenko, and Vadim Fomin. 2018. Conditional random fields for metaphor detection. In *Proceedings of the Workshop on Figurative Language Processing*, New Orleans,LA.

Agnieszka Mykowiecka, Aleksander Wawer, and Malgorzata Marciniak. 2018. Detecting figurative word occurrences using word embeddings. In *Proceedings of the Workshop on Figurative Language Processing*, New Orleans,LA.

Yair Neuman, Dan Assaf, Yohai Cohen, Mark Last, Shlomo Argamon, Newton Howard, and Ophir Frieder. 2013. Metaphor identification in large texts corpora. *PloS one*, 8(4):e62343.

Gözde Özbal, Carlo Strapparava, and Serra Sinem Tekiroglu. 2016. Prometheus: A corpus of proverbs annotated with metaphors. In *LREC*.

Malay Pramanick, Ashim Gupta, and Pabitra Mitra. 2018. An lstm-crf based approach to token-level metaphor detection. In *Proceedings of the Workshop on Figurative Language Processing*, New Orleans,LA.

Marek Rei, Luana Bulat, Douwe Kiela, and Ekaterina Shutova. 2017. Grasping the finer point: A supervised similarity network for metaphor detection. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1537–1546.

Ekaterina Shutova, Douwe Kiela, and Jean Maillard. 2016. Black holes and white rabbits: Metaphor identification with visual features. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 160–170.

Ekaterina Shutova, Lin Sun, Elkin Darío Gutiérrez, Patricia Lichtenstein, and Srini Narayanan. 2017. Multilingual metaphor processing: Experiments with semi-supervised and unsupervised learning. *Computational Linguistics*, 43(1):71–123.

Ekaterina Shutova, Lin Sun, and Anna Korhonen. 2010. Metaphor identification using verb and noun clustering. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1002–1010. Association for Computational Linguistics.

Filip Skurniak, Maria Janicka, and Aleksander Wawer. 2018. Multim-module recurrent neural networks with transfer learning. a submission for the metaphor detection shared task. In *Proceedings of the Workshop on Figurative Language Processing*, New Orleans,LA.

Gerard J Steen, Aletta G Dorst, J Berenike Herrmann, Anna Kaal, Tina Krennmayr, and Trijntje Pasma. 2010. *A method for linguistic metaphor identification: From MIP to MIPVU*, volume 14. John Benjamins Publishing.

Egon Stemle and Alexander Onysko. 2018. Using language learner data for metaphor detection. In *Proceedings of the Workshop on Figurative Language Processing*, New Orleans,LA.

Krishnkant Swarnkar and Anil Kumar Singh. 2018. Di-lstm contrast : A deep neural network for metaphor detection. In *Proceedings of the Workshop on Figurative Language Processing*, New Orleans,LA.

Serra Sinem Tekiroglu, Gözde Özbal, and Carlo Strapparava. 2015. Exploring sensorial features for metaphor identification. In *Proceedings of the Third Workshop on Metaphor in NLP*, pages 31–39.

Paul H Thibodeau and Lera Boroditsky. 2011. Metaphors we think with: The role of metaphor in reasoning. *PloS one*, 6(2):e16782.

Yulia Tsvetkov, Leonid Boytsov, Anatole Gershman, Eric Nyberg, and Chris Dyer. 2014. Metaphor detection with cross-lingual model transfer. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 248–258.

Yulia Tsvetkov, Elena Mukomel, and Anatole Gershman. 2013. Cross-lingual metaphor detection using common semantic features. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 45–51.

Peter D Turney, Yair Neuman, Dan Assaf, and Yohai Cohen. 2011. Literal and metaphorical sense identification through concrete and abstract context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 680–690. Association for Computational Linguistics.

Tony Veale, Ekaterina Shutova, and Beata Beigman Klebanov. 2016. Metaphor: A computational perspective. *Synthesis Lectures on Human Language Technologies*, 9(1):1–160.

Chuhan Wu, Fangzhao Wu, Yubo Chen, Sixing Wu, Zhigang Yuan, and Yongfeng Huang. 2018. Thu ngn at naacl-2018 metaphor shared task: Neural metaphor detecting with cnn-lstm model. In *Proceedings of the Workshop on Figurative Language Processing*, New Orleans,LA.

Gerald Zaltman and Lindsay H Zaltman. 2008. *Marketing metaphoria: What deep metaphors reveal about the minds of consumers*. Harvard Business Press.