Learning the Peculiar Value of Actions

Daniel Dahlmeier
Research & Innovation, SAP Asia, Singapore
`d.dahlmeier@sap.com`

Abstract

We consider the task of automatically estimating the value of human actions. We cast the problem as a supervised learning-to-rank problem between pairs of action descriptions. We present a large, novel data set for this task which consists of challenges from the I Will If You Will Earth Hour challenge. We show that an SVM ranking model with simple linguistic features can accurately predict the relative value of actions.

1 Introduction

The question on how humans conceptualize value is of great interest to researchers in various fields, including linguistics (Jackendoff, 2006). The link between value and language arises from the fact that we cannot directly observe value due to its abstract nature and instead often study language expressions that describe actions which have some value attached to them. This creates an interesting link between the semantics of the words that describe the actions and the underlying moral value of the actions.

Jackendoff (2006) describes value as an "internal accounting system" for ethical decision processes that exhibits both *valence* (good or bad) and *magnitude* (better or worse). Most interestingly, value is governed by a "peculiar logic" that provides constraints on which actions are deemed morally acceptable and which are not. In particular, the principal of *reciprocity* states that the valence and magnitude of reciprocal actions (actions that are done "in return" for something else) should match, i.e., positive valued actions should

match with positive valued reciprocal actions (reactions) of similar magnitude, and conversely negatively valued actions should match with negative valued reciprocal actions (reactions) of similar magnitude.

In this paper, we consider the task of automatically estimating the value of actions. We present a simple and effective method for learning the value of actions from ranked pairs of textual action descriptions based on a statistical learning-to-rank approach. Our experiments are based on a novel data set that we create from challenges submitted to the I Will if You Will Earth Hour challenge where participants pledge to do something daring or challenging if other people commit to sustainable actions for the planet. Our method achieves a surprisingly high accuracy of up to 94.72% in a 10-fold cross-validation experiment. The results show that the value of actions can accurately be estimated by machine learning methods based on lexical descriptions of the actions.

The main contribution of this paper is that we show how the semantics of value in language can accurately be learned from empirical data using a learning-to-rank approach. Our work shows an interesting link between empirical research on semantics in natural language processing and the concept of value.

2 The Logic of Value

Our approach is based on the concept of value as presented by Jackendoff (2006) who describes value as an abstract property that is attributed to objects, persons, and actions. He further describes logical inference rules that humans use to determine which actions are deemed morally acceptable and which are not. The most important inference rule for our work is the principal of *reciprocation*, things that are done "in return" for some other action (Fiengo and Lasnik, 1973). In English, this relation is often expressed by the prepo-

sition *for*, as shown by the following example sentences (Jackendoff, 2006).

1. Susan praised Sam *for* behaving nicely.
2. Fred cooked Lois dinner *for* fixing his computer.
3. Susan insulted Sam *for* behaving badly.
4. Lois slashed Fred's tires *for* insulting her.

The first two examples describe actions with positive value, while the last two examples describe actions with negative value. We expect that the valence values of reciprocal actions match: positively valued actions demand a positively valued action in return, while negatively valued actions trigger negatively valued responses. If we switch the example sentences and match positive actions with negative actions, we get sentences that sound counter-intuitive or perhaps comical (we prefix counter-intuitive sentences with a hash character '#').

1. #Susan insulted Sam for behaving nicely.
2. #Lois slashed Fred's tires for fixing her computer.

Similarly, we expect that the magnitudes of value between reciprocal actions match. Sentences where the magnitude of the value of the response action does not match the magnitude of the initial action seem odd or socially inappropriate (overacting/underacting).

1. #Fred cooked Lois dinner for saying hello to him.
2. #Fred cooked Lois dinner for rescuing all his relatives from certain death.
3. #Fred slashed Lois's tires for eating too little at dinner.
4. #Fred slashed Lois's tires for murdering his entire family.

We observe that reciprocal actions typically match each other in valence and magnitude. Coming back to our initial goal of learning the value of actions, this gives us a method for comparing the value of actions that were done in return to the same initial action.

3 I Will If You Will challenge

The I Will If You Will (IWIYW) challenge[1] is part of the World Wildlife Fund's Earth Hour campaign

| I will quit smoking if you will start recycling. (500 people) |
| I will adopt a panda if you will start recycling. (1000 people) |
| I will dance gangnam style if you will plant a tree. (100 people) |
| I will dye my hair red if you will upload an IWIYW challenge. (500 people) |
| I will learn Java if you will upload an IWIYW challenge. (10,000 people) |

Table 1: Examples of I Will If You Will challenges.

which has the goal to increase awareness of sustainability issues. In this challenge, participants make a pledge to do something daring or challenging if a certain number of people commit to sustainable actions for the planet. The challenges are created by ordinary people on the Earth Hour campaign website. Each challenge takes the form of a simple school yard dare: *I will do X, if you will do Y*, where *X* is typically some daring or challenging task that the challenge creator commits to do if a sufficient number of people commit to do action *Y* which is some sustainable action for the planet. Together with the textual description, each challenge includes the number of people that need to commit to doing *Y* in order for the challenge creator to perform *X*. Examples of the challenges are shown in Table 1.

It is important to note that during the challenge creation on the IWIYW website, the *X* challenge is a free text input field that allows the author to come up with creative and interesting challenges. The sustainable actions *Y* and the number of people that need to commit to it are usually chosen from a fixed list of choices. As a result, there is a large number of different *X* actions and a comparably smaller number of *Y* actions. The collected challenges provide a unique data set that allows us to quantitatively measure the value of each promised task by the number of people that need to fulfill the sustainable action.

4 Method

In this section, we present our approach for estimating the value of actions. Our approach casts the problem as a supervised learning-to-rank problem between pairs of actions. Given, a textual description of an action a, we want to estimate its

value magnitude v. We represent the action a via a set of features that are extracted from the description of the action. We use a linear model that combines the features into a single scalar value for the value v

$$v = w^T \mathbf{x^a}, \quad (1)$$

where $\mathbf{x^a}$ is the feature vector for action description a and w is a learned weight vector. The goal is to learn a suitable weight vector w that approximates the true relationship between textual expressions of actions and their magnitude of value.

Instead of estimating the value directly, we take an alternative approach and consider the task of learning the relative ranking of pairs of actions. We follow the pairwise approach to ranking (Herbrich et al., 1999; Cao et al., 2007) that reduces ranking to a binary classification problem. Ranking the values v_1 and v_2 of two actions a_1 and a_2 is equivalent to determining the sign of the dot product between the weight vector w and the difference between the feature vectors $\mathbf{x^{a_1}}$ and $\mathbf{x^{a_2}}$.

$$
\begin{aligned}
v_1 > v_2 &\Leftrightarrow w^T \mathbf{x^{a_1}} > w^T \mathbf{x^{a_2}} \\
&\Leftrightarrow w^T \mathbf{x^{a_1}} - w^T \mathbf{x^{a_2}} > 0 \\
&\Leftrightarrow w^T (\mathbf{x^{a_1}} - \mathbf{x^{a_2}}) > 0 \quad (2)
\end{aligned}
$$

For each ranking pair of actions, we create two complimentary classification instances: $(\mathbf{x^{a_1}} - \mathbf{x^{a_2}}, l_1)$ and $(\mathbf{x^{a_2}} - \mathbf{x^{a_1}}, l_2)$, where the labels are $l_1 = +1, l_2 = -1$ if the first challenge has higher value than the second challenge and $l_1 = -1, l_2 = +1$ otherwise. We can train a standard linear classifier on the generated training instances to learn the weight vector w.

In the case of the IWIYW data, there is no explicit ranking between actions. However, we are able to create ranking pairs for the IWIYW data in the following way. As we have seen, there is only a small set of different *You Will* challenges that are reciprocal actions for a diverse set of *I Will* challenges. Thus, many *I Will* challenges will end up having the same *You Will* challenge. We can use the *You Will* challenges as a pivot to effectively "join" the *I Will* challenges. The number of required people to perform *Y* induces a natural ordering between the values of the *I Will* actions where a higher number of required participants means that the *I Will* task has higher value.

For example, for the challenges displayed in Table 1, we can use the common *You Will* challenges

to create the following ranked challenge pairs.

I will quit smoking < I will adopt a panda

I will dye my hair red < I will learn Java   (3)

According to the examples, adopting a panda has higher value than quitting smoking and learning Java has higher value than dying ones hair red. The third challenge does not share a common *You Will* challenge with any other challenge and therefore no ranking pairs can be formed with it.

As the IWIYW challenges are created online in a non-controlled environment, we have to expect that there is some noise in the automatically created ranked challenges. However, a robust learning algorithm has to be able to handle a certain amount of noise. We note that our method is not limited to the IWIYW data set but can be applied to any data set of actions where relative rankings are provided or can be induced.

4.1 Features

The choice of appropriate feature representations is crucial to the success of any machine learning method. We start by parsing each *I Will If You Will* challenge with a constituency parser. Because each challenge has the same *I Will If You Will* structure, it is easy to identify the subtrees that correspond to the *I Will* and *You Will* parts of the challenge. An example parse tree of a challenge is shown in Figure 1. The yield of the *You Will* subtree serves as a pivot to join different *I Will* challenges. To represent the *I Will* action a as a feature vector $\mathbf{x^a}$, we extract the following lexical and syntax features from the *I Will* subtree of the sentence.

- **Verb:** We extract the verb of the *I Will* clause as a feature. To identify the verb, we pick the left-most verb of the *I Will* subtree based on its part-of-speech (POS) tag. We extract the lowercased word token as a feature. For example, for the sentence in Figure 1, the verb feature is *verb=quit*. If the verb is negated (the left sibling of the *I Will* subtree spans exactly the word *not*), we add the postfix *NOT* to the verb feature, for example *verb=quit_NOT*.

- **Object:** We take the right sibling of the *I will* verb as the object of the action. If the right sibling is a particle with constituent label PRT, e.g., *travel **around** the UK on bike*,
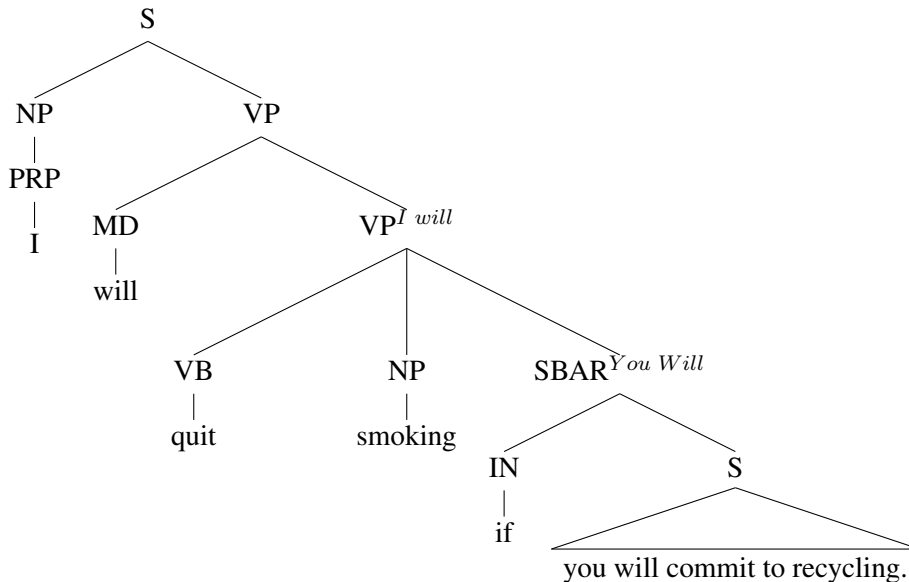
Figure 1: Parse tree of a *I Will If You Will* challenge. The subtrees governing the *I Will* and *You Will* part of the sentence are marked.

we skip the particle and take the next sibling as the object. If the object is a prepositional phrase with constituent tag PP, e.g., *go **without** electricity for a month*, we take the second child of the prepositional phrase as the object phrase. We then extract two features to represent the object. First, we extract the lowercased head word of the object as a feature. Second, we extract the concatenation of all the words in the yield of the object node as a single feature to capture the complete argument for longer objects. In our example sentence, the object head feature and the complete object feature are identical: *object_head=smoking* and *object=smoking*.

- **Unigram:** We take all lowercased words that are not stopwords in the *I Will* part of the sentence as binary features. In our example sentence, the unigram features *unigr_quit* and *unigr_smoking* would be active.

- **Bigram:** We take all lowercased bigrams in the *I Will* part of the sentence as binary features. We do not remove stopwords for bigram features. In our example sentence, the bigram features *bigr_quit_smoking* would be active.

We note that our method is not restricted to these feature templates. More sophisticated features, like tree kernels (Collins and Duffy, 2002) or se-

mantic role labeling (Palmer et al., 2010), can be imagined.

5 Experiments

We evaluate our approach using standard 10-fold cross-validation and report macro-average accuracy scores for each of the feature sets. The classifier in all our experiments is a linear SVM implemented in SVM-light (Joachims, 2006).

5.1 Data

We obtained a snapshot of 18,290 challenges created during the 2013 IWIYW challenge. The snapshot was taken in mid May 2013, just 1.5 weeks before the 2013 Earth Hour event day. We perform the following pre-processing. We normalize the text to proper UTF-8 encoding and remove challenges where the complete sentence contained less than 7 tokens. These challenges were usually empty or incomplete. We filter the challenges using the `langid.py` tool (Lui and Baldwin, 2012) and only keep English challenges. We normalized the casing of the sentences by first lowercasing all texts and then re-casing each sentence with a simple re-casing model that replaces a word with its most frequent casing form. The re-casing model is trained on the Brown corpus (Ku and Francis, 1967). We tokenize the sentences with the Penn Treebank tokenizer. We parse the sentences with the Stanford parser (Klein and Manning, 2003a; Klein and Manning, 2003b) to ob-

Features	Accuracy
random	0.5000
verb	0.6241
unigrams	0.8481
unigrams + verb	0.8573
object	0.8904
verb + object	0.9115
bigrams	0.9251
unigrams + bigrams	0.9343
unigrams + bigrams + verb	0.9361
unigrams + bigrams + verb + object	**0.9472**

Table 2: Results of 10-fold cross-validation experiments.

tain a constituency parse tree for each challenge. After pre-processing, we are left with 5,499 challenges (4,982 unique), with 4,474 unique *I Will* challenges and 70 unique *You Will* challenges.

We create binary classifications examples between pairs of actions as described in Section 4. As we create all possible combinations between *I Will* challenges with common *You Will* challenges, the number of ranking pairs for training is large. In our case, we ended up with over 840,000 classification instances. We note that not every *I Will* action is guaranteed to be included in the final set of ranking pairs as challenges with a unique *You Will* part that is not found in any other challenge cannot be joined and are effectively ignored. However, this is not a problem for our experiments. The binary classification instances are used to train and test a ranking model for estimating the value of actions as described in the last section.

5.2 Results

The results of our cross-validation experiments are shown in Table 2.

The random baseline for all experiments is 50%. Just using the verb of the *I Will* action as a feature improves over the random baseline to 62.41%. Using a unigram bag-of-words representation of the actions achieves a very respectable score of 84.81%. When we combine unigrams with the verb feature, we achieve 85.73%. One of the most surprising results of our experiments is that the object of the action alone is a very effective feature, achieving 89.04%. When combined with the verb feature, the object feature achieves 91.15% which shows that the verb and object carry most of the relevant information that the model requires

to gauge the value of actions. Using bigrams as features, seems to catch this information just as accurately, achieving 92.51% accuracy. The score is further improved by combining the different feature sets. The best result of 94.72% is obtained by combining all the features: unigrams, bigrams, verb, and object. In summary, these results show that our method is able to accurately predict the relative value of actions using simple linguistic features, which is the main contribution of this work.

6 Related Work

The concept of value and reciprocity has been extensively studied in the social sciences (Gergen and Greenberg, 1980), anthropology (Sahlins, 1972), economics (Fehr and Gächter, 2000), and philosophy (Becker, 1990). In linguistics, value has been studied by Jackendoff (2006). His work forms the starting point of our approach.

In natural language processing, there has been very little work on the concept of value. Paul *et al.* (2009) and Girju and Paul (2011) address the problem of semi-automatically mining patterns that encode reciprocal relationships using pronoun templates. Their work focuses on mining patterns of reciprocity while our work uses expressions of reciprocal actions to learn the value of actions.

None of the above works tries to estimate the value of actions, as we do in this work. In fact, we are not aware of any other work that tries to estimate the value of actions from lexical expressions of value.

7 Conclusion

We have presented a simple and effective method for learning the value of actions from reciprocal sentences. We show that our SVM-based ranking model with simple linguistic features is able to accurately rank pairs of actions from the I Will If You Will Earth Hour challenge, achieving an accuracy of up to 94.72%.

Acknowledgement

References

Lawrence C Becker, editor. 1990. *Reciprocity*. University of Chicago Press.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning (ICML)*, pages 129–136.

Michael Collins and Nigel Duffy. 2002. Convolution kernels for natural language. In *Advances in Neural Information Processing Systems 14 (NIPS 2001)*, pages 625–632.

Ernst Fehr and Simon Gächter. 2000. Cooperation and punishment in public goods experiments. pages 980–994.

Robert Fiengo and Howard Lasnik. 1973. The logical structure of reciprocal sentences in English. *Foundations of language*, pages 447–468.

Kenneth J. Gergen and Willis Richard H. Greenberg, Martin S., editors. 1980. *Social exchange: Advances in theory and research*. Plenum Press.

Roxana Girju and Michael J Paul. 2011. Modeling reciprocity in social interactions with probabilistic latent space models. *Natural Language Engineering*, 17(1):1–36.

Ralf Herbrich, Thore Graepel, and Klaus Obermayer. 1999. Support vector learning for ordinal regression. In *In Proceedings of the 1999 International Conference on Articial Neural Networks*, pages 97–102.

Ray Jackendoff. 2006. The peculiar logic of value. *Journal of Cognition and Culture*, 6(3-4):375–407.

Thorsten Joachims. 2006. Training linear SVMs in linear time. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 217–226.

Dan Klein and Christopher D. Manning. 2003a. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL 2003)*, pages 423–430.

Dan Klein and Christopher D. Manning. 2003b. Fast exact inference with a factored model for natural language parsing. *Advances in Neural Information Processing Systems 15 (NIPS 2002)*, pages 423–430.

Henry Ku and W. Nelson Francis. 1967. *Computational Analysis of Present-Day American English*. Brown University Press.

Marco Lui and Timothy Baldwin. 2012. An off-the-shelf language identification tool. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL 2012)*, pages 25–30.

Martha Palmer, Daniel Gildea, and Nianwen Xue. 2010. Semantic role labeling. *Synthesis Lectures on Human Language Technologies*, 3(1):1–103.

Michael Paul, Roxana Girju, and Chen Li. 2009. Mining the web for reciprocal relationships. In *Proceedings of the 13th Conference on Computational Natural Language Learning (CoNLL)*, pages 75–83.

Marshall D. Sahlins. 1972. *Stone age economics*. Transaction Publishers.