Agree or Disagree: Predicting Judgments on Nuanced Assertions

Michael Wojatzki **Torsten Zesch**
Language Technology Lab
University of Duisburg-Essen, Germany
michael.wojatzki@uni-due.de
torsten.zeschi@uni-due.de

Saif M. Mohammad **Svetlana Kiritchenko**
National Research Council Canada
Ottawa, Canada
saif.mohammad@nrc-cnrc.gc.ca
svetlana.kiritchenko@nrc-cnrc.gc.ca

Abstract

Being able to predict whether people agree or disagree with an assertion (i.e. an explicit, self-contained statement) has several applications ranging from predicting how many people will like or dislike a social media post to classifying posts based on whether they are in accordance with a particular point of view. We formalize this as two NLP tasks: predicting judgments of (i) individuals and (ii) groups based on the text of the assertion and previous judgments. We evaluate a wide range of approaches on a crowdsourced data set containing over 100,000 judgments on over 2,000 assertions. We find that predicting individual judgments is a hard task with our best results only slightly exceeding a majority baseline. Judgments of groups, however, can be more reliably predicted using a Siamese neural network, which outperforms all other approaches by a wide margin.

1 Introduction

One of the most basic reactions when reading a sentence is to agree or disagree with it.[1] Mechanisms that allow us to express agreement (e.g. thumb-up, like, up-vote, ♡) or disagreement (e.g. thumb-down, dislike, down-vote) towards posts of other users can be found in almost all social networking sites. The judgments associated with posts that discuss controversial political or social issues, such as *legalization of drug*, *immigration policy*, or *gun rights*, are a rich source of information for those interested in the opinions of individuals or groups. For instance, public opinion regarding an issue is often illustrated by the number of retweets, likes, or upvotes that a politician or influential person receives.

[1] You are probably thinking about whether you agree with that statement right now.

Hence, especially for controversial issues, being able to predict how people judge posts has several applications: *people at large* could automatically anticipate if politicians, companies or other decision makers would agree or disagree with a new perspective on a problem or how they would evaluate a new possible solution. The method can also be used by *journalists* to more accurately analyze the homogeneity of opinions or to detect filter bubbles in social media. *Decision makers* themselves would be able to evaluate in advance how citizens, customers, or employees react to a press announcement, a new regulation, or tweet. *Social media users* could be enabled to search, sort or filter posts based on whether they are in accordance with or contrary to their personal world view. Such predictions could also be used to augment chat applications by indicating to a user if her recipients will agree or disagree with a message to be sent, enabling to choose a more or less confrontational discussion style.

In this paper, we describe how the outlined use cases can be framed as two inference tasks: predicting individual judgments and predicting judgments of whole groups. As a first step, we restrict ourselves to judgments on textual utterances that are explicit, relevant, and that do not contain multiple positions. We will refer to such utterances as *assertions*. For solving the tasks, we define the degree to which two assertions are judged similar as **judgment similarity**. This similarity allows us to predict a judgment based on other judgments that have been made on similar, known assertions.

Across both tasks, we compare this strategy against several baselines and reference approaches on a newly crowdsourced data set containing over 100 000 judgments on assertions. We find that, for predicting individual judgments, our best results only slightly exceed a majority baseline, but that judgments of groups can be more reliably pre-

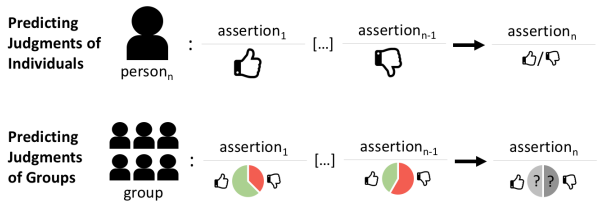


Figure 1: Overview on the two prediction tasks.

dicted using a Siamese neural network, which outperforms all other approaches by a wide margin.

2 Predicting Judgments

In order to predict if someone will agree with an assertion, we need knowledge about that person. Ideally, we would have access to a large set of other assertions which the person has already judged. We could then measure the similarity between previous assertions and the new assertion and hypothesize that the judgment on the new assertion should be the same as for a highly similar one. In Figure 1, we show this case for binary (yes/no) predictions on individuals and argue that this can be also generalized to probabilistic predictions on groups of people. Thus, we formulate two prediction tasks:

In the first task, we want to **predict judgments of individuals** on assertions based on other judgments by the same person. Thus, the first task is formulated as follows: given a set of assertions $a_1, ..., a_n$ relevant to an issue and the judgments of a person p_i on $a_1, ..., a_{n-1}$ an automatic system has to predict p_i's judgment on the assertion a_n.

In the second task, we want to **predict judgments of groups** on assertions based on averaged judgments of other assertions. Hence, this task can be formalized as follows: given a set of judgments of a group of persons $p_1, ..., p_k$ on the assertions $a_1, ..., a_{n-1}$, an automatic systems must predict the judgment on the assertion a_n for the same group of persons. Judgments of groups can be expressed by an aggregated agreement score between -1 and 1, where -1 means that every person disagrees to an assertion and 1 that every person agrees to the assertion.

For measuring the similarity between two assertions, we propose to compare how a large group of people judges them. We define the degree to which two assertions are judged similarly by a large group as the **judgment similarity** of the two assertions. However, judgments of other persons are not easily available – e.g. if we want to predict a judgment on a new, unseen assertion. To overcome this limitation, we propose to use methods that consider the texts of the assertions to mimic judgment similarity and have thus the ability to generalize from existing data collections.

3 Related Work

Measuring the judgment similarity of two assertions is related to several NLP tasks such as the detection of semantic text similarity (STS) (Agirre et al., 2012), paraphrase recognition (Bhagat and Hovy, 2013), and textual entailment (Dagan et al., 2009).

Unlike semantic text similarity, we do not use a notation of similarity based on the intuition of humans, but one that derives from the context of judgments. Hence, we define that the judgment similarity of two assertions is 1 if two assertions are consistently judged the same and are thus interchangeable in the context of our task.

There are several reasons why assertions are judged similarly: their text may convey similar semantics such as in the assertions '*Marijuana alleviates the suffering of chronically ill patients*' and '*Marijuana helps chronically ill persons*'. This type of similarity corresponds to what methods of semantic text similarity capture. However, a strong judgment similarity of two assertions can also be due to semantically entailed relationships between assertions. For instance, if people agree with '*Marijuana is a gateway drug for teenagers and damages growing brains*' most of them also agree to '*Marijuana is dangerous for minors*', despite the texts being different in content and having thus low semantic text similarity. In addition, two assertions can also have a strong judgment similarity because of underlying socio-cultural, political, or personal factors. For instance, the assertions '*Consuming Marijuana has no impact on your success at work*' and '*Marijuana is not addictive*' describe different arguments for legalizing marijuana, but judgments made on these assertions are often correlated.

Our work also relates to other attempts on predicting reactions to text, such as predicting the number of retweets (Suh et al., 2010; Petrovic et al., 2011), the number of likes on tweets (Tan et al., 2014), the number of karma points of reddit posts (Wei et al., 2016), or sales from product descriptions (Pryzant et al., 2017). What those

works have in common is that they measure some kind of popularity, which differs significantly from our task: even if one agrees with a text, one might decide not to retweet or like it for any number of reasons. There are also cases in which one may retweet a post with which one disagrees in order to flag someone or something from the opposing community. Furthermore, there are effects such as the author's followers affecting the visibility of posts and thereby the likelihood of a like or a retweet (Suh et al., 2010).

In addition, we relate to works that aim at predicting whether two texts (Menini and Tonelli, 2016) or sequences of utterances (Wang and Cardie, 2014; Celli et al., 2016) express agreement or disagreement with each other. More broadly, we also relate to works that analyze stance (Mohammad et al., 2016; Xu et al., 2016; Taulé et al., 2017), sentiment (Pang and Lee, 2008; Liu, 2012; Mohammad, 2016), or arguments (Habernal and Gurevych, 2016; Boltuzic and Šnajder, 2016; Bar-Haim et al., 2017) that are expressed via text. In contrast to these works, we do not examine what judgment, sentiment, or claim is expressed by a text, but whether we can infer agreement or disagreement based on judgments which were made on other assertions.

Finally, we relate to work on analyzing and predicting outcomes of congressional roll-call voting. These works constantly find that votes of politicians can be explained by a low number of underlying, ideological dimensions such as being left or right (Heckman and Snyder, 1996; Poole and Rosenthal, 1997, 2001). Our work is different from these attempts, as we do not consider politicians who might have incentives to vote in accordance with the ideological views of their party, and as we base our prediction on the text of assertions.

4 Data Collection

For exploring how well the two tasks can be solved automatically, we use the dataset *Nuanced Assertions on Controversial Issues (NAoCI)* created by Wojatzki et al. (2018). The dataset contains assertions judged on a wide range of controversial issues.[2] The NAoCI dataset mimics a common situation in many social media sites, where people e.g. up- or downvote social media posts. However, it does not have the experimental problems of using social media data directly. These problems include legal reasons of scraping social media data and moderator variables such as the definition of issues, the influence of previous posts, or the question of whether someone is not judging an assertion because she does not want to judge it or because she did not perceive it.

The data was collected using crowdsourcing conducted on `crowdflower.com` in two steps. First, participants were asked to generate a large set of assertions relevant to controversial issues. The set of assertions was created using crowdsourcing, as a manual creation of assertions would be potentially incomplete and subject to personal bias. We provided instructions to make sure that the assertions are natural, self-contained statements about an issue. Next, a large number of people was asked to indicate whether they agree or disagree with these assertions.

The process was reviewed and approved by the institutional ethics board of the National Research Council Canada.

Generating Assertions In order to obtain realistic assertions, 69 participants were asked to generate assertions for sixteen predefined issues (see Table 1). For each issue, the subjects were given definition of the issue and a few example assertions. In addition, the instructions state that assertions should be explicit, relevant to an issue, self-contained, and only contain a single position. Specifically, the use of co-reference or hedging indicated by words such as *perhaps*, *maybe*, or *possibly* was not permitted. After a removal of duplicates and instances that did not follow the rules, this process resulted in about 150 unique assertions per issue (2,243 in total).

Judging Assertions Next, 230 subjects were asked to indicate whether they agree or disagree with an assertion, resulting in over 100 000 judgments (see Table 1). The participants were free to judge as many assertions on as many issues as they wanted. On average each assertion is judged by about 45 persons and each participant judged over 400 assertions. For each person, agreement is encoded with 1, disagreement with -1, and missing values with 0 (as not all subjects judged all assertions). Additionally, we can also compute the *aggregated agreement score* for each assertion by simply subtracting the percentage of participants that disagreed with the assertion from the

[2]The dataset is accessible from `https://sites.google.com/view/you-on-issues/`

Issue	# of Assertions	# of Judgments
Black Lives Matter	135	6 154
Climate Change	142	6 473
Creationism in School	129	5 747
Foreign Aid	150	6 866
Gender Equality	130	5 969
Gun Rights	145	6 423
Marijuana	138	6 200
Mandatory Vaccination	134	5 962
Media Bias	133	5 877
Obama Care	154	6 940
Same-sex Marriage	148	6 899
US Electoral System	175	7 695
US in the Middle East	138	6 280
US Immigration	130	5 950
Vegetarian & Vegan Lifestyle	128	5 806
War on Terrorism	134	5 892
Total	2 243	101 133

Table 1: Issues and number of crowdsourced assertions and judgments.

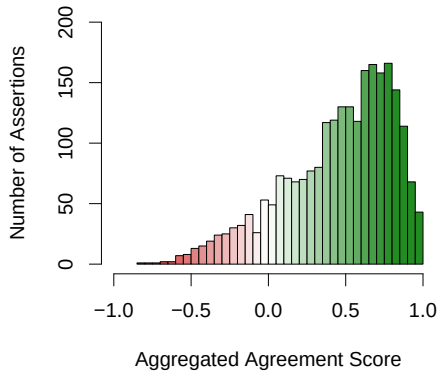


Figure 2: Distribution of aggregated agreement scores.

percentage of participants that agreed with the assertion. Figure 2 shows the distribution of aggregated agreement scores (grouped into bins of size .05) across all issues. The mass of the distribution is concentrated in the positive range of possible values, which indicates that the participants more often agree with the assertions than they disagree. Consequently, baselines accounting for this imbalance perform strongly in predicting judgments on assertions. However, the distribution corresponds to what we observe in many social network sites, where e.g. the ratio of likes to dislikes is also clearly skewed towards likes.

All data, the used questionnaires along with the directions and examples are publicly available on the project website.[2]

5 Measuring Judgment Similarity Between Assertions

As mentioned above, we want to predict judgments on a previously unseen assertion based on judgments of similar assertions. For that purpose, we need to measure the similarity of assertions $sim(a_1, a_2)$ based on their text only. For measuring the similarity of two assertions we rely on the judgment matrix J, with $j_{p,a}$ as the judgment provided by participant p for assertion a, with $\vec{j_p}$ as the row vector of all ratings of participant p, and $\vec{j_a}$ as the column vector of all ratings provided for assertion a. We measure the gold similarity of two assertions by comparing their judgment vectors in the matrix. If the vectors are orthogonal, the assertions are maximally dissimilar (i.e. persons who agree to assertion a_1 disagree with a_2). If the vectors are parallel, the assertions have a perfect similarity. We compute the cosine similarity between the judgment vectors of two assertions. We calculate the gold similarity between all unique pairs (e.g. we do not use both a_1 with a_2 and a_2 with a_1) in our data and do not consider self-pairing.

5.1 Experimental Setup

As baselines for this task, we utilize well-established semantic text similarity (STS) methods that calculate overlap between the surface forms of assertions. We use the following methods as implemented by DKPro Similarity (Bär et al., 2013)[3]: (i) unigram overlap expressed by the Jaccard coefficient (Lyon et al., 2001), (ii) greedy string tiling (Wise, 1996), (iii) longest common sub string (Gusfield, 1997). Additionally, we use averaged word embeddings (Bojanowski et al., 2017).

Beyond the baselines, we apply two machine learning approaches: a conventional SVM-based classifier and a neural network. The SVM classifier is implemented using LibSVM (Chang and Lin, 2011) as provided by DKProTC (Daxenberger et al., 2014).[4] We use a combination of various ngram features, sentiment features (derived from the system by Kiritchenko et al. (2014)[5]), embedding features (averaged embeddings by Bojanowski et al. (2017)) and negation features. We used a linear kernel with C=100 and the nu-SVR

[3] version 2.2.0

[4] version 1.0

[5] The NRC-Canada system ranked first in the SemEval 2013 (Nakov et al., 2013) and 2014 (Rosenthal et al., 2014) tasks on sentiment analysis.

regression model. Iterative experiments showed that this configuration gave the most stable results across the issues. For the neural approach, we adapt Siamese neural networks (SNN), which consist of two identical branches or sub-networks that try to extract useful representations of the assertions and a final layer that merges these branches. SNNs have been successfully used to predict text similarity (Mueller and Thyagarajan, 2016; Neculoiu et al., 2016) and match pairs of sentences (e.g. a tweet to reply) (Hu et al., 2014). In our SNN, a branch consists of a layer that translates the assertions into sequences of word embeddings, which is followed by a convolution layer with a filter size of two, max pooling over time layer, and a dense layer. To merge the branches, we calculate the cosine similarity of the extracted vector representations. The SNN was implemented using the deep learning framework deepTC (Horsmann and Zesch, 2018) in conjunction with Keras[6] and Tensorflow (Abadi et al., 2016). In order to ensure full reproducibility of our results, the source code for both approaches is publicly available.[7] We evaluate all approaches using 10-fold cross validation and calculate Pearson correlation between the prediction and the gold similarity.

5.2 Results

Table 2 shows the correlation of all approaches averaged over all sixteen issues.[8] Overall, the STS baselines result in very low correlation coefficients between .02 and .07, while the trained models obtain coefficients around .6. This shows that the systems can learn useful representations that capture judgment similarity and that this representation is indeed different from semantic similarity. Since both models are purely lexical and still yield reliable performance, we suspect that the relationship between a pair of assertions and their judgment similarity also has a lexical nature.

While STS baselines obtain consistently low results, we observe largely differing results per issues (ranging from .32 to .72) with SVM and SNN behaving alike. Detailed results for each issue are listed in Table 5 the appendix.

In order to better understand the results, we examine the scatter-plots that visualize assignment of gold to prediction (x–Axis: gold, y–Axis: prediction) and investigate cases that deviate strongly from an ideal correlation. Figure 3 shows the scatter plot for the issue *Climate Change* for both classifiers. For the SVM we observe that there is a group of pairs that is predicted inversely proportional, i.e. their gold value is positive, but the regression assigns a clearly negative value. We observe that these instances mainly correspond to pairs in which both assertions have high negative word scores. For instance the pair, '*There is not a real contribution of human activities in Climate change*' and '*Climate change was made up by the government to keep people in fear*', have a comparable high similarity of .20. The SVM, however, assigns them a similarity score of $-.38$. We suspect that this effect results from the distribution of similarity scores that is skewed to the positive range of possible scores. Therefore, the SVM probably assigns too much weight to ngrams that signal a negative score. Far less pronounced, for the neural approach, we find instances whose gold values are negative, but which are assigned a positive value. When inspecting these pairs we find that many of them contain one assertion which uses a negation (e.g. *not*, *unsure*, or *unlikely*). An example for this is the pair, '*There has been an increase in tropical storms of greater intensity which can be attributed to climate change*' and '*Different changes in weather does not mean global warming*', that have a low similarity in the gold data (-0.19), but get assigned a rather high similarity score (.20).

Method	r
SNN	.61
SVM	.58
Embedding distance	.07
Jaccard	.07
Greedy string tiling	.06
Longest common sub string	.05

Table 2: Pearson correlation (averaged over all issues) of text-based approaches for approximating similarity of assertion judgments.[5]

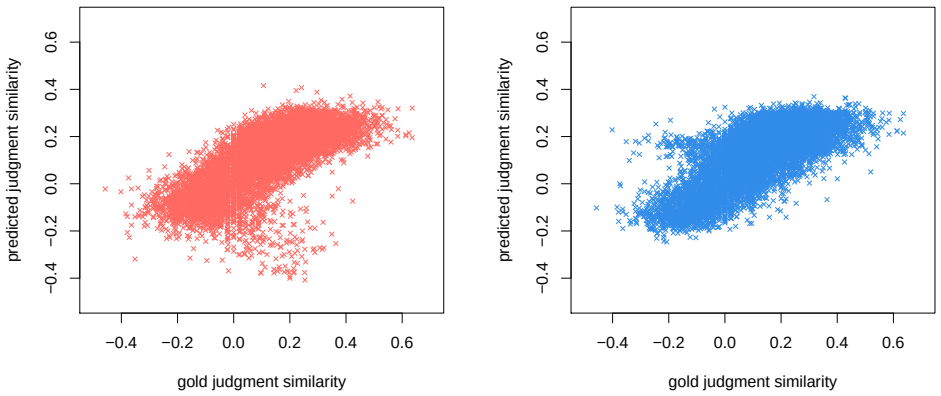
6 Predicting Judgments of Individuals

Now that we have means for quite reliably estimating the judgment similarity of assertions, we can try to predict judgments on individual assertions. We compare the judgment similarity meth-

[6] https://keras.io/

[7] https://github.com/muchafel/judgmentPrediction

[8] As Pearsons r is defined in a probabilistic space it cannot be averaged directly. Therefore, we first z-transform the scores, average them and then transform them back into the original range of values.



(a) SVM (b) SNN

Figure 3: Comparison of gold judgment similarity and judgment similarity as predicted by the SVM and the SNN for the issue *Climate Change*.

ods against several baselines and collaborative filtering methods (that make use of judgments that are made by other persons to calculate person and assertion similarity).

Baselines (BL) The **random** baseline predicts *agree* or *disagree* for each assertion. We also define the **all agree** baseline, which always predicts *agree*. As the data contains substantially more agree judgments than disagree judgments (c.f. Figure 2), this is a strong baseline. As a third baseline, we average all known judgments of a person and predict *agree* if this value is positive and predict *disagree* otherwise. We refer to this baseline as **tendency**.

Judgment Similarity (JS) We use the above defined judgment similarity methods to calculate the similarity between each of the assertions previously judged by that person and the assertion for which we want to make the prediction. Then we simply transfer the judgment of the most similar assertion to the assertion of interest.[9] To prevent leakage, the scores of the prediction are taken from the models that have been trained in the cross validation. This means, for predicting the score of a pair of assertions we use the model which does not include the pair in the training set. As the matrix is missing one entry for each prediction (i.e. the judgment on the assertion for which we want to make the prediction), one could theoretically form a new matrix for each prediction and then re-calculate all cosines. However, we find that the judgment similarity between assertions does not change significantly when a single entry in the vectors of the assertions is removed or added. Hence, due to computational complexity, the gold similarity was calculated over the entire matrix of judgments.

There are several assertions that do not have textual overlap, which is why the STS methods often return a zero similarity. In such a case, we fall back on the *all agree* baseline. We refer to strategies which are based on judgment similarity as **most similar assertion (*method*)**, where *method* indicates how the similarity is computed.

All strategies use all available context. For instance, if we want to predict the judgment of the assertion a_n and a prediction strategy considers other judgments, the strategy uses all the judgments on the assertions $a_1, ..., a_{n-1}$.

Collaborative Filtering (CF) Collaborative filtering (Adomavicius and Tuzhilin, 2005; Schafer et al., 2007; Su and Khoshgoftaar, 2009) uses previously made judgments and judgments made by others to predict future judgments. Collaborative filtering has been successfully used in application areas such shopping recommendations (Linden et al., 2003), or personalization of news (Das et al., 2007). Note that collaborative filtering requires knowledge of how others judged the assertion for which the system tries to make a predic-

[9] Note that the subjects have all rated different numbers of assertions. Thus, for the sake of comparability, we restrict ourselves to the most similar assertion (as opposed to averaging a judgment over the n most similar assertions.)

Strategy	Type	Accuracy
most similar user	CF	.85
most similar assertion (gold)	CF	.76
tendency	BL	.75
mean other	CF	.74
most similar assertion (SNN)	JS	.73
most similar assertion (SVM)	JS	.72
all agree	BL	.71
most similar assertion (jaccard)	JS	.70
most similar assertion (embedding)	JS	.68
most similar assertion (gst)	JS	.69
most similar assertion (lcss)	JS	.67
random	BL	.50

Table 3: Accuracy of different approaches for predicting judgments of individuals.

tion. Therefore, these strategies are not applicable if we want to predict judgments on a previously unseen assertion. Nevertheless, they represent an upper bound for our text-based predictions.

As a simple collaborative filtering strategy, we predict how the majority of other persons judged an assertion. Therefore, we average the judgments of all other users and predict *agree* if this value is positive and *disagree* if the value is negative. This strategy will be referred to as **mean other**. In addition, we compute the similarity between pairs of people by calculating the cosine similarity between the vector that corresponds to all judgments a person has made. We use this person–person similarity to determine the most similar person and then transfer the judgment on a_n of the user which is most similar to p_i. We refer to this strategy as **most similar user**. We also use the (gold) judgment similarity between assertions to predict *agree* or *disagree* based on how the assertion that is most similar to a_n has been judged. We call this strategy **most similar assertion (gold)**.

6.1 Results

Table 3 shows the accuracy of the strategies across all issues obtained using leave-one-out cross validation. We observe that all strategies are significantly better than the random baseline. On average, the *all agree* strategy is more than 20% above the random baseline and thus represents a highly competitive baseline. The *tendency* baseline, which is a refinement of *all agree*, is even 4% higher. Only the collaborative filtering strategies *most similar assertion* and **most similar user** beat this baseline. With an accuracy of about 85% the *most similar user* strategy performs best. The methods that use the learned judgment similarity beat the *all agree* but fall behind the *tendency* baseline. The fact that methods based on judgment similarity are already close to their upper-bound (**most similar assertion (gold)**) shows that their potential is limited, even if measuring judgment similarity can be significantly improved. One possible explanation for comparably low performance of *most similar assertion* is that the past assertions are not sufficient to make a meaningful prediction. For instance, if only a few assertions have been judged in the past and none of them is similar to a new assertion, then a prediction becomes guessing. As expected from their poor performance of approximating judgment similarity, the methods relying on STS measures fall behind the *all agree*. In the appendic (Table 6) we show how the strategies perform on the individual issues.

7 Predicting Judgments of Groups

We now turn to predicting judgments of groups, i.e. the task of estimating what percentage of a group of people are likely to agree to an assertion. We illustrate the prediction task in the following example: From the assertion '*Marijuana is almost never addictive*' with an aggregated agreement score of 0.9 we want to predict a comparatively lower value for the assertion '*Marijuana is sometimes addictive*'.

Direct Prediction (DP) As a reference approach, we train different regression models that predict the aggregated agreement score directly from the text of the assertion. We train each model over all issues in order to achieve the necessary generalization.

Again, we compare more traditional models based on feature engineering and neural models. For the feature engineering approach we experiment with the following feature sets: First, we use a length feature which consists of the number of words per assertion. To capture stylistic variations, we compute a feature vector consisting of the number of exclamation and question marks, the number of modal verbs, the average word length in an assertion, POS type ratio, and type token ratio. We capture the wording of assertions by different ngram features. For capturing the semantics of words, we again derive features from the pre-trained fastText word vectors (Bojanowski et al., 2017). To capture the emotional tone of an assertion, we extract features from the output of the readily available sen-

Model	Type	r
gold ($n = 7$)	JS	.90
gold ($n = 1$)	JS	.84
SNN ($n = 34$)	JS	.74
SNN ($n = 1$)	JS	.45
SVM ($n = 18$)	JS	.42
CNN	DP	.40
sentiment + trigrams	DP	.36
trigrams	DP	.35
unigrams + embeddings	DP	.32
unigrams	DP	.32
SVM ($n = 1$)	JS	.32
sentiment + trigrams + style	DP	.27
sentiment	DP	.13
style	DP	.10
length	DP	.00

Table 4: Correlation coefficients for approaches on predicting judgments of groups.

timent tool NRC-Canada Sentiment Analysis System (Kiritchenko et al., 2014).

As the neural approach on directly predicting aggregated judgments, we use a single branch of the Siamese network. However, since we are trying to solve a regression problem, here the network ends in a single node equipped with a linear activation function. Through iterative experiments we found out that it is advantageous to add two additional dense layers before the final node. As this model resembles a convolutional neural network (CNN), we label this approach as CNN.

Judgment Similarity (JS) In analogy to the prediction of judgments of individuals, we first calculate the judgment similarity of two assertions using the SVM and SNN approaches that take pair of assertions into account. We then take the n-most similar assertions and return the average of the resulting scores. As an upper bound, we also compute the judgment similarity that results from the gold data. Note, that this upper bound again assumes knowledge about judgments on the assertion for which we actually want to make a prediction. We make the code for both approaches publicly available.[10]

7.1 Results

Table 4 shows the performance of the different approaches for predicting judgments of groups. For the prediction based on judgment similarity, we observe large differences between the the SVM and SNN predictions. This is especially interesting because the performance of the similarity prediction is comparable. We attribute this to the systematic error made by the SVM when trying to predict the similarity of assertions that have a negative agreement score. While the SVM only outperforms the plain regressions if the prediction is based on several assertions, we observe a substantially better performance for the judgment similarity based on the SNN. For the best judgment similarity model (SNN with $n = 34$), we obtain a coefficient of $r = .74$ which is substantially better than the direct prediction model (CNN, $r = .40$).

For the plain regression, we observe that the CNN outperforms all models based on feature engineering and that among the SVM models ngram features yield the best performance. While the sentiment feature alone has low performance, the model that combines sentiment and ngrams shows slight improvement over the trigrams alone. The length feature and the style features alone have a comparable low performance and models which combine these feature with lexical features show a lower performance than the lexical models alone.

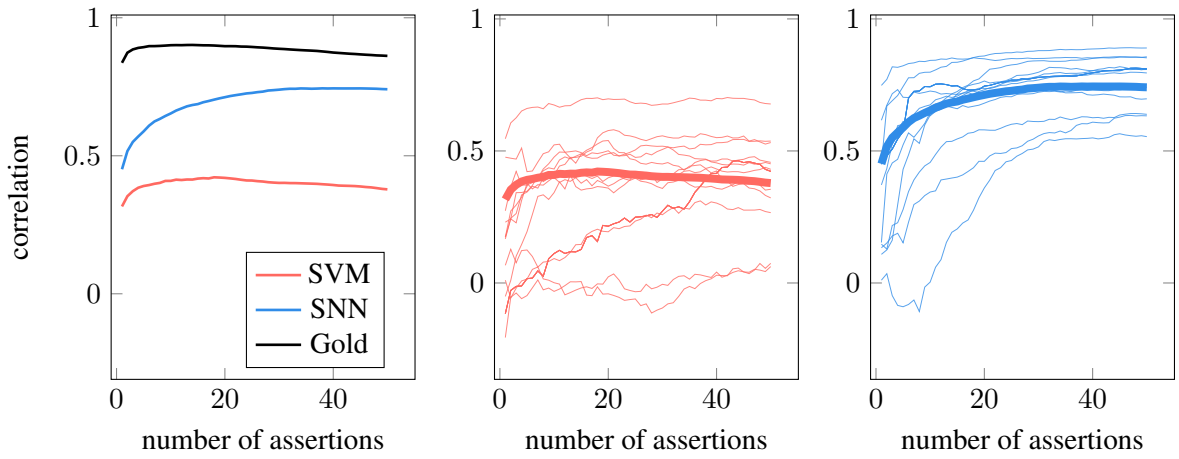
Issue-wise analysis To better understand the differences between the judgment similarity methods, we inspect their performance depending on the number of given assertions. Figure 4 shows this comparison both for individual issues and averaged across all issues. The upper-bound reaches a correlation of up to $r = .89$ ($n = 8$). The strength of this correlation and the fact that even our best estimate is still 15 points less shows the potential of judgment similarity for predicting judgments of groups.

For the SNN, the predictions follow a similar pattern: resembling a learning curve, the performance increases rapidly with increasing n, but then plateaus from a certain number of assertions. However, the number of assertions for which we observe a plateau varies significantly. For the SVM we observe a similar pattern for most of the approaches, but the plateau is often reached much later. There are two issues (*US Engagement in the Middle East* and *US Immigration*) where we do not observe an increase in performance with increasing n. We suspect that the systematic error of the SVM is particularly strong here.

8 Conclusion & Future Work

In this paper, we examined whether an automatically measured judgment similarity can be used to predict the judgments of individuals or groups on

[10] https://github.com/muchafel/judgmentPrediction



(a) Averaged (b) issue-wise (SVM) (c) issue-wise (SNN)

Figure 4: Prediction quality based on the transfer of the n-most similar assertions (expressed by the strength of correlation with the gold values). Sub-figure a) shows the scores averaged across all issues. We show the variance obtained on individual issues by the SVM in Sub-Figure b) and by the SNN in Sub-Figure c).

assertions. We compare these judgment similarity approaches against several reference approaches on a data set of over 100,000 judgments on over 2,000 assertions. For the prediction of individual judgments reference approaches yield competitive results. However, for the prediction of group judgments the best approach using judgment similarity as predicted by a SNN outperforms other approaches by a wide margin.

While the presented approaches represent a first take on predicting judgments on assertions, the proposed tasks also suggest several directions of future research. These include more advanced algorithmic solutions and experiments for obtaining a deeper understanding of the relationship between text and judgments. For improving the automatic prediction, we want to explore how robust the learned models are by examining whether they can be transferred between issues. In addition, we want to examine if knowledge bases, issue specific corpora, or issue specific word vectors can improve the current approaches. To better understand what textual properties of assertions cause judgment similarity, we want to annotate and experimentally control typed relationships (e.g. paraphrases, entailment) of pairs of assertions. Being able to predict the degree to which two assertions are judged similarly might also be helpful for NLP tasks in which one tries to predict opinions or stance of the author of an text. Hence, we want to examine if judgment similarity can be used to boost the performance of systems in these tasks.

Acknowledgments

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under grant No. GRK 2167, Research Training Group "User-Centred Social Media". We would also like to thank Tobias Horsmann for helpful discussions on implementing the SNN.

References

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2016. TensorFlow: A System for Large-scale Machine Learning. In *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, OSDI'16, pages 265–283, Savannah, USA.

Gediminas Adomavicius and Alexander Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749.

Eneko Agirre, Mona Diab, Daniel Cer, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of the SemEval*, pages 385–393, Montreal, Canada.

Daniel Bär, Torsten Zesch, and Iryna Gurevych. 2013. DKPro Similarity: An Open Source Framework for Text Similarity. In *Proceedings of the 51st Annual*

Meeting of the Association for Computational Linguistics: System Demonstrations, pages 121–126, Sofia, Bulgaria.

Roy Bar-Haim, Indrajit Bhattacharya, Francesco Dinuzzo, Amrita Saha, and Noam Slonim. 2017. Stance Classification of Context-Dependent Claims. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL*, pages 251–261, Valencia, Spain.

Rahul Bhagat and Eduard Hovy. 2013. What is a paraphrase? *Computational Linguistics*, 39(3):463–472.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Filip Boltuzic and Jan Šnajder. 2016. Fill the gap! analyzing implicit premises between claims from online debates. In *Proceedings of the Third Workshop on Argument Mining (ArgMining2016)*, pages 124–133, Berlin, Germany.

Fabio Celli, Evgeny Stepanov, Massimo Poesio, and Giuseppe Riccardi. 2016. Predicting Brexit: Classifying agreement is better than sentiment and pollsters. In *Proceedings of the Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media (PEOPLES)*, pages 110–118, Osaka, Japan.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Ido Dagan, Bill Dolan, Bernardo Magnini, and Dan Roth. 2009. Recognizing textual entailment: Rational, evaluation and approaches. *Natural Language Engineering*, 15(4):i–xvii.

Abhinandan S. Das, Mayur Datar, Ashutosh Garg, and Shyam Rajaram. 2007. Google News Personalization: Scalable Online Collaborative Filtering. In *Proceedings of the 16th International Conference on World Wide Web (WWW '07)*, pages 271–280, Banff, Canada.

Johannes Daxenberger, Oliver Ferschke, Iryna Gurevych, and Torsten Zesch. 2014. DKPro TC: A Java-based Framework for Supervised Learning Experiments on Textual Data. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 61–66, Baltimore, Maryland.

Dan Gusfield. 1997. *Algorithms on strings, trees and sequences: computer science and computational biology*. Cambridge university press.

Ivan Habernal and Iryna Gurevych. 2016. Which Argument is More Convincing? Analyzing and Predicting Convincingness of Web Arguments Using Bidirectional LSTM. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1589–1599, Berlin, Germany.

James J. Heckman and James M. Snyder. 1996. Linear Probability Models of the Demand for Attributes with an Empirical Application to Estimating the Preferences of Legislators. *The RAND Journal of Economics*, 28:142–189.

Tobias Horsmann and Torsten Zesch. 2018. DeepTC – An Extension of DKPro Text Classification for Fostering Reproducibility of Deep Learning Experiments. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, Miyazaki, Japan.

Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional Neural Network Architectures for Matching Natural Language Sentences. In *Advances in Neural Information Processing Systems (NIPS 27)*, pages 2042–2050, Montreal, Canada.

Svetlana Kiritchenko, Xiaodan Zhu, and Saif M Mohammad. 2014. Sentiment Analysis of Short Informal Texts. *Journal of Artificial Intelligence Research (JAIR)*, 50:723–762.

Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon.com recommendations: Item-to-item Collaborative Filtering. *IEEE Internet computing*, 7(1):76–80.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Caroline Lyon, James Malcolm, and Bob Dickerson. 2001. Detecting short passages of similar text in large document collections. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, pages 118–125, Pittsburgh, USA.

Stefano Menini and Sara Tonelli. 2016. Agreement and Disagreement: Comparison of Points of View in the Political Domain. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*, pages 2461–2470, Osaka, Japan.

Saif M. Mohammad. 2016. Sentiment analysis: Detecting valence, emotions, and other affectual states from text. *Emotion Measurement*, pages 201–238.

Saif M. Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. SemEval-2016 Task 6: Detecting Stance in Tweets. In *Proceedings of the SemEval*, pages 31–41, San Diego, USA.

Jonas Mueller and Aditya Thyagarajan. 2016. Siamese Recurrent Architectures for Learning Sentence Similarity. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, pages 2786–2792, Phoenix, USA.

Preslav Nakov, Sara Rosenthal, Zornitsa Kozareva, Veselin Stoyanov, Alan Ritter, and Theresa Wilson. 2013. SemEval-2013 Task 2: Sentiment Analysis in Twitter. In *Proceedings of the SemEval*, pages 312–320, Atlanta, USA.

Paul Neculoiu, Maarten Versteegh, and Mihai Rotaru. 2016. Learning text similarity with siamese recurrent networks. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 148–157.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135.

Sasa Petrovic, Miles Osborne, and Victor Lavrenko. 2011. RT to Win! Predicting Message Propagation in Twitter. *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, 11:586–589.

Keith T. Poole and Howard Rosenthal. 1997. *Congress: A Political-economic History of Roll Call Voting*. Oxford University Press.

Keith T. Poole and Howard Rosenthal. 2001. D-Nominate after 10 Years: A Comparative Update to Congress: A Political-Economic History of Roll-Call Voting. *Legislative Studies Quarterly*, 26(1):5–29.

Reid Pryzant, Young-joo Chung, and Dan Jurafsky. 2017. Predicting Sales from the Language of Product Descriptions. In *Proceedings of the SIGIR 2017 Workshop on eCommerce (ECOM 17)*, Tokyo, Japan.

Sara Rosenthal, Alan Ritter, Preslav Nakov, and Veselin Stoyanov. 2014. Semeval-2014 task 9: Sentiment analysis in twitter. In *Proceedings of the SemEval*, pages 73–80, Dublin, Ireland.

J. Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. 2007. Collaborative Filtering Recommender Systems. *The Adaptive Web*, pages 291–324.

Xiaoyuan Su and Taghi M. Khoshgoftaar. 2009. A Survey of Collaborative Filtering Techniques. *Advances in Artificial Intelligence*, pages 4:2–4:2.

Bongwon Suh, Lichan Hong, Peter Pirolli, and Ed H. Chi. 2010. Want to be Retweeted? Large Scale Analytics on Factors Impacting Retweet in Twitter Network. In *Proceedings of the Second IEEE International Conference on Social Computing*, pages 177–184, Washington, USA.

Chenhao Tan, Lillian Lee, and Bo Pang. 2014. The Effect of Wording on Message Propagation: Topic-and Author-controlled Natural Experiments on Twitter. In *Proceedings of the ACL*, pages 175–185, Baltimore, USA.

Mariona Taulé, M Antonia Martı, Francisco Rangel, Paolo Rosso, Cristina Bosco, and Viviana Patti. 2017. Overview of the task of stance and gender detection in tweets on catalan independence at ibereval 2017. In *Notebook Papers of 2nd SEPLN Workshop on Evaluation of Human Language Technologies for Iberian Languages (IBEREVAL), Murcia, Spain, September*, volume 19.

Lu Wang and Claire Cardie. 2014. Improving Agreement and Disagreement Identification in Online Discussions with A Socially-Tuned Sentiment Lexicon. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA '14)*, pages 97–106.

Zhongyu Wei, Yang Liu, and Yi Li. 2016. Is This Post Persuasive? Ranking Argumentative Comments in the Online Forum. In *Proceedings of the ACL*, pages 195–200, Berlin, Germany.

Michael J. Wise. 1996. Yap3: Improved detection of similarities in computer program and other texts. *ACM SIGCSE Bulletin*, 28(1):130–134.

Michael Wojatzki, Saif M. Mohammad, Torsten Zesch, and Svetlana Kiritchenko. 2018. Quantifying Qualitative Data for Understanding Controversial Issues. In *Proceedings of the 11th Edition of the Language Resources and Evaluation Conference (LREC-2018)*, Miyazaki, Japan.

Ruifeng Xu, Yu Zhou, Dongyin Wu, Lin Gui, Jiachen Du, and Yun Xue. 2016. Overview of NLPCC Shared Task 4: Stance Detection in Chinese Microblogs. In *International Conference on Computer Processing of Oriental Languages*, pages 907–916, Kunming, China.

A Appendix

Issue	SVM	SNN
Climate Change	.70	.72
Gender Equality	.67	.73
Mandatory Vaccination	.68	.74
Obama Care	.66	.70
Black Lives Matter	.66	.74
Media Bias	.63	.63
US Electoral System	.63	.59
Same-sex Marriage	.59	.61
War on Terrorism	.56	.59
Foreign Aid	.54	.46
US in the Middle East	.52	.55
US Immigration	.52	.57
Gun Rights	.51	.64
Creationism in school	.48	.51
Vegetarian and Vegan Lifestyle	.43	.40
Legalization of Marijuana	.37	.32

Table 5: Correlation coefficients of the similarity prediction by the SVM and the SNN, obtained in 10 fold cross-validation.

Strategy	Type	Average	Black Lives Matter	Climate Change	Creationism in school	Foreign Aid	Gender Equality	Gun Rights	Marijuana	Same-sex Marriage	Mandatory Vaccination	Media Bias	Obama Care	US Electoral System	US in the Middle East	US Immigration	Vegetarianism & Veganism	War on Terrorism
most similar user	CF	**.85**	**.86**	**.86**	**.85**	**.83**	**.87**	**.86**	**.85**	**.86**	**.87**	**.85**	**.85**	**.82**	**.84**	**.85**	**.83**	**.84**
most similar assertion	CF	**.76**	**.78**	**.79**	**.73**	**.74**	**.79**	**.75**	**.71**	**.80**	**.80**	**.77**	**.77**	**.74**	**.73**	**.74**	**.74**	**.72**
tendency	BL	**.75**	**.77**	**.79**	**.71**	**.76**	**.78**	**.71**	**.68**	**.70**	**.78**	**.79**	**.77**	**.76**	**.75**	**.75**	**.68**	**.74**
mean other	CF	**.74**	**.79**	**.80**	**.70**	**.74**	**.78**	**.75**	.63	**.66**	**.77**	**.77**	**.75**	**.72**	**.73**	**.75**	**.70**	**.73**
most similar assertion (SNN)	JS	**.73**	**.75**	**.78**	.67	.73	.75	**.75**	**.66**	**.75**	**.76**	.75	**.76**	.70	.70	.70	**.69**	.70
most similar assertion (SVM)	JS	**.72**	.74	**.77**	**.68**	.72	.74	.69	**.64**	**.72**	**.76**	.75	**.74**	**.72**	.70	.70	**.68**	.69
all agree	BL	.71	.75	.77	.68	.73	.76	.69	.64	.64	.75	.77	.74	.72	.72	.73	.62	.71
most similar assertion (jaccard)	JS	.70	.74	.73	.67	.71	.74	.69	.62	**.66**	**.76**	.77	.73	.72	.68	.71	**.63**	.70
most similar assertion (embedding)	JS	.68	.68	.70	.65	.67	.74	.63	.62	**.67**	.70	.72	.67	.69	.67	.67	**.66**	.69
most similar assertion (gst)	JS	.69	.69	.72	.65	.68	.73	.66	.64	**.68**	.73	.72	.67	.71	.68	.67	**.63**	.69
most similar assertion (lcss)	JS	.67	.69	.72	.63	.66	.70	.65	.59	**.66**	.65	.71	.71	.70	.67	.67	**.63**	.66
random	BL	.50	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5	.5

Table 6: Accuracy of different prediction approaches. Results above the *all agree* baseline are boldfaced.

Table 5 shows the correlation coefficients of the similarity prediction by the SVM and the SNN per issue.

Table 6 shows the performance of the different strategies on predicting judgments of individuals per issue. On Average the *tendency* baseline and the *all agree* baseline turned out to be extremely competitive. However, for the issues *Marijuana* and *Vegetarism & Veganism* both baselines are outperfomed by the learned judgment similarity (both SVM and SNN). In addition, the STS methods outperform the *all agree* baseline on theses issues.