From Surrogacy to Adoption; From Bitcoin to Cryptocurrency: Debate Topic Expansion

Roy Bar-Haim*, **Dalia Krieger***, **Orith Toledo-Ronen***, **Lilach Edelstein, Yonatan Bilu, Alon Halfon, Yoav Katz, Amir Menczel, Ranit Aharonov and Noam Slonim**
IBM Research

Abstract

When debating a controversial topic, it is often desirable to expand the boundaries of discussion. For example, we may consider the pros and cons of possible alternatives to the debate topic, make generalizations, or give specific examples. We introduce the task of *Debate Topic Expansion* - finding such related topics for a given debate topic, along with a novel annotated dataset for the task. We focus on relations between Wikipedia concepts, and show that they differ from well-studied lexical-semantic relations such as hypernyms, hyponyms and antonyms. We present algorithms for finding both consistent and contrastive expansions and demonstrate their effectiveness empirically. We suggest that debate topic expansion may have various use cases in argumentation mining.

1 Introduction

Recent years saw substantial advancement of *Debating Technologies* – computational technologies developed directly to enhance, support, and engage with human debating (Gurevych et al., 2016). A recent milestone in this field is IBM®'s *Project Debater*®[1], the first demonstration of a live competitive debate between an AI system and a human debate champion.

When debating a controversial topic, it is often desirable to expand the boundaries of the discussion, and bring up arguments about related topics.

For example, when discussing the pros and cons of the *presidential system*, it is natural to contrast it with those of the *parliamentary system*. When debating *alternative medicine*, we may discuss specific examples, such as *homeopathy* and *naturopathy*. Conversely, when discussing *bitcoins*, we can speak more broadly on *cryptocurrency*.

Consider the use of debating technologies for decision support, where the pros and cons of a given proposal are extracted from a large corpus, summarized and presented to the user. Current methods for topic-related, corpus-wide argument mining only specify the given debate topic in their search queries (Levy et al., 2017, 2018; Wachsmuth et al., 2017; Stab et al., 2018a,b). As a result, much of the relevant argumentative content is left out of their reach. Alternatively, context-independent argument mining can exhaustively extract argumentative content from a corpus (Lippi and Torroni, 2015), but it cannot tell which arguments are actually relevant for the topic in question.

In this work we take a step towards closing this gap, by introducing the task of *Debate Topic Expansion* – finding related topics that can enrich our arguments and strengthen our case when debating a given topic. Following previous work (Levy et al., 2017, 2018), we focus on topics that are Wikipedia concepts (article titles in Wikipedia).

Two types of expansions are studied: *consistent* and *contrastive* (Bar-Haim et al., 2017). Arguing in favor or against a consistent expansion may support the *same* stance towards the original topic, whereas for contrastive expansions the stance is reversed. For example, *Bitcoin⇒Cryptocurrency* and *Alternative medicine⇒Homeopathy* are consistent expansions, while *Presidential system⇒Parliamentary system* is a contrastive expansion, since we may support the presidential system by criticizing the parliamentary system. While these relations may seem reminiscent of hypernyms/hyponyms, antonyms and co-hyponyms, we show that they differ from these well-studied relations.

We propose a three-step method for debate topic expansion. First, expansion candidates are extracted from a large corpus using a set of prede-

*First three authors equally contributed to this work.

[1] https://www.research.ibm.com/artificial-intelligence/project-debater/

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 977–990
Florence, Italy, July 28 - August 2, 2019.

fined patterns. Each expansion type makes use of a different type of corpus and patterns. In the second step, we apply a set of filters to the extraction results. Candidates that pass these filters are manually annotated as good/bad expansions, resulting in the first dataset for this task.[2]

The labeled dataset is utilized in the final step, where we employ supervised classification to identify good expansions amongst the candidates. We explore two approaches: (i) traditional feature-based classification, for which we introduce a novel set of features; (ii) a deep neural network, which is trained by distant supervision. Experiments with hundreds of unseen topics show promising results, and the best performance is achieved by combining both classification approaches.

2 Task Description

Let DC (*debate concept*) be a Wikipedia concept representing a debate topic. Our goal is to find Wikipedia concepts that represent consistent and contrastive expansions of DC, as defined in the previous section. Table 1 lists several positive and negative examples of candidates extracted for each expansion type, denoted $\Rightarrow$ and $\not\Rightarrow$, respectively. The examples are taken from our labeled dataset, to be described in the next sections.

Consistent expansions in our dataset include both broader concepts (examples 1,8,10 in Table 1) and more specific concepts (examples 3,5,7). While some of these expansions are strict hypernyms (1,8) or hyponyms (5,7), other are not (3,10). Moreover, broader/narrower concepts do not necessarily make relevant expansions. For example, while *Vegetarianism* is a type of *Diet* (2), arguing about diets in general does not seem relevant for a debate about vegetarianism, in particular since such a debate typically contrasts vegetarianism with other types of diet.

Contrastive expansions involve diverse semantic relations and subtle distinctions. For example, *opposites* may be relevant expansions in some cases, e.g. (19), but irrelevant in others (15). Contrastive expansions are often co-hyponyms of the DC. For instance, *Democracy* and *Dictatorship* (11) are both forms of government. However, co-hyponyms are not always appropriate as contrastive expansions. When debating about *Boxing* (e.g., whether it should be banned), we would not contrast it with *Wrestling* (17), despite both being combat sports, as most arguments for and against boxing equally apply to wrestling.

The above examples illustrate some of the challenges in this task. The criterion for a good expansion – its usefulness in a debate – requires some knowledge and understanding of the possible contexts in which the given DC may be debated. Moreover, such judgments are, to some extent, inherently subjective.

3 Acquisition of Expansion Candidates

3.1 Candidate Extraction

The first step in expanding a given DC is extracting *expansion concepts* for each expansion type. An expansion concept (EC) is a Wikipedia concept that co-occurs with the DC in some predefined pattern that is matched in a corpus. We use a wikification tool that matches different variations of mentioning the same concept in the text. Below we describe the patterns and corpora used for each expansion type. For both types we require at least two pattern matches for each expansion concept.

Consistent expansions. Our list of patterns for extracting consistent expansions includes some of the well-known Hearst patterns for extracting hyponyms (Hearst, 1992), as well as some additional patterns. Some examples are 'X *such as* Y', 'X *is a* Y', and 'X *and other* Y', where X matches DC and Y matches EC *or* vice versa.[3] Despite the differences between hypernyms/hyponyms and consistent expansions, these patterns provide a reasonable starting point for our algorithm. The patterns are matched in a corpus $\mathcal{C}_N$ of news articles, comprising about 10 billion sentences. The sentences undergo wikification and indexing, which allows efficient pattern search for a given concept.

Contrastive expansions. As previously observed by Bar-Haim et al. (2017), queries to web search engines often contain contrastive expressions, e.g. *"why is* ***renewable energy*** *better than* ***fossil fuels****"*, and are typically succinct and easy to parse.[4] We used a corpus $\mathcal{C}_Q$ of 1.2 billion queries

[2] The dataset is available at https://www.research.ibm.com/haifa/dept/vst/debating_data.shtml

[3] See appendices B.1 and B.2 for a complete list of consistent and contrastive patterns.

[4] It is unlikely, however, to find a concept and its generalization in the same query, hence we did not use this corpus for extracting consistent expansions.

	Consistent Expansions				Contrastive Expansions		
	Debate Concept		Expansion Concept		Debate Concept		Expansion Concept
1	Wind power	$\Rightarrow$	Renewable energy	11	Democracy	$\Rightarrow$	Dictatorship
2	Vegetarianism	$\not\Rightarrow$	Diet (nutrition)	12	Democracy	$\not\Rightarrow$	Socialism
3	Entrepreneurship	$\Rightarrow$	Startup company	13	Flat tax	$\Rightarrow$	Progressive tax
4	Abortion	$\not\Rightarrow$	Fetus	14	Flat tax	$\not\Rightarrow$	Value-added tax
5	Fast food	$\Rightarrow$	McDonald's	15	Global warming	$\not\Rightarrow$	Global cooling
6	Fast food	$\not\Rightarrow$	Restaurant	16	Carbon tax	$\Rightarrow$	Emission trading
7	Gender inequality	$\Rightarrow$	Gender pay gap	17	Boxing	$\not\Rightarrow$	Wrestling
8	Gender inequality	$\Rightarrow$	Social inequality	18	Polyamory	$\Rightarrow$	Monogamy
9	Organic food	$\not\Rightarrow$	Local food	19	Private university	$\Rightarrow$	Public university
10	Casino	$\Rightarrow$	Gambling	20	Recycling	$\Rightarrow$	Landfill

Table 1: Positive ($\Rightarrow$) and negative ($\not\Rightarrow$) examples of consistent and contrastive expansions.

(450 million distinct queries) from the Blekko search engine. Sample patterns include 'X * *vs* * Y', 'X * *better than* * Y', and '*difference between* * X * *and* * Y', where '*' matches any number of non-concept tokens.

3.2 Candidate Filtering

Candidate extraction is followed by a candidate filtering step, in which we apply a set of filters to each extracted pair (DC,EC). The filters for each expansion type are described below.

Filters for Consistent Expansions

Directionality. We aim to determine whether a consistent expansion EC is a generalization or a specialization of the DC based on the number of times EC it is matched in a each role in any of the patterns. If the direction is not clearly determined, i.e. the majority role is matched in less than 80% of the cases, the expansion is discarded.

Named Entity. Only the more specific concept amongst DC and EC (according to the determined direction) may be a named entity[5].

Frequency Ratio. Incompatible frequencies of DC and EC may indicate a bad expansion. Accordingly, this filter restricts the ratio between the frequencies of DC and EC in $\mathcal{C}_N$. We require $min(\frac{Freq(DC)}{Freq(EC)}, \frac{Freq(EC)}{Freq(DC)}) \geq 0.2$.

Distributional Similarity. Consistent expansions are expected to occur in contexts similar to the DC. This is captured by measuring the distributional similarity s_D between DC and EC. We derived concept-level word2vec vectors (Mikolov et al., 2013a) from $\mathcal{C}_N$, where each wikified mention of a concept C was considered an occurrence of C. $s_D(DC, EC)$ is then defined as the cosine similarity between the representations of DC and EC, and we require $s_D(DC, EC) \geq 0.5$. This filter may also remove EC that is too broad or too narrow with respect to DC.

Substring. We require that EC is not a substring of DC and vice versa. This filter discards expansions such as *Private university*$\not\Rightarrow$*University* and *Marriage*$\not\Rightarrow$*Gay Marriage*.

Additional filters. We also filter concepts containing the phrases 'Anti-', 'List of' and 'Lists of', and pairs (DC,EC) that co-occur in the same sentence in $\mathcal{C}_N$ less than 10 times.

Filters for Contrastive Expansions

Named Entity. Neither DC nor EC may be named entities.

Substring. Same as for consistent expansions.

Semantic relatedness. We found that unlike consistent relations, contrastive relations better correlate with semantic relatedness than with distributional similarity. We use *WORT* (Ein Dor et al., 2018), a semantic relatedness tool for Wikipedia concepts, as our relatedness measure. We denote this relation s_R, and require that $s_R \geq 0.4$.

4 The Debate Topic Expansion Dataset

Based on our candidate acquisition method, we created the *DTE* (Debate Topic Expansion) dataset, comprising about 3,000 annotated pairs of debate concepts and their expansion candidates. The dataset contains positive and negative examples of both consistent and contrastive expansions. The construction of the dataset is described below.

[5] A Wikipedia concept is considered a named entity if its page type is defined as person, organization or location.

We manually collected a diverse set of 632 debate concepts from a variety of sources, including the idebate website[6]. For each debate concept, we performed candidate extraction and filtering for consistent and contrastive expansions, as described in the previous section. Each of the resulting (DC,EC) pairs was assessed by five annotators, and was labeled as either positive (good expansion) or negative (bad expansion), based on the majority labeling.

One intriguing subtlety we noticed early on is that in the case of contrastive expansions, whether or not EC is a good expansion somewhat depends on our stance towards the DC. If we argue *against* the DC (*Con* stance), we may choose any plausible alternative as our EC, following a line of argument such as "EC *is a better alternative to* DC". However, when we argue *in favor of* DC (*Pro* stance), the typical argument changes to "if we don't choose DC then we are left with EC", which requires EC to be the "default" alternative to DC. For example, when arguing against atheism, one may argue that Christians are happier than atheists; however, when taking a pro-atheism stance, it is better to argue against religion in general than specifically against Christianity. The annotators were therefore asked to assess contrastive expansions for both positive and negative stances. However, developing a classifier that is able to make such fine distinctions falls out of the scope of the current work. Instead, we take the union of good expansions for each stance as our positive instances, while keeping per-stance annotations in the dataset for future research.

Table 2 provides some statistics on the resulting dataset. Our candidate acquisition method was found to be applicable to a significant portion of the topics: one or more good consistent expansions were found for 43% of the topics, and good contrastive expansions were found for 19% of the topics. Precision, however, is low, even after applying our filters: 49% for consistent expansions, and 19% for contrastive expansions. This motivates an additional supervised classification step, to be presented in the next section. These statistics suggest that identifying contrastive expansions is considerably more challenging than finding consistent expansions.

Fleiss' κ is 0.45 for consistent expansions and 0.43 for the unified contrastive expansions, which corresponds to "moderate agreement" (Landis and Koch, 1997). This level of agreement reflects the complexity and inherent subjectivity of the task, as discussed in Section 2, and is comparable to previous results for annotation tasks in argumentation mining. For example, Aharoni et al. (2014) report κ of 0.39-0.4 for claim and evidence annotation in Wikipedia articles.

	Consistent Expansions		Contrastive Expansions	
Debate topics				
Total	632		632	
With expansions	360	(57%)	286	(45%)
With good expansions	269	(43%)	120	(19%)
Annotated expansions				
Total	1,741		1,326	
Good expansions	845	(49%)	251	(19%)
Inter-annotator agreement				
Fleiss' κ	0.45		0.43	

Table 2: Statistics on the DTE (Debate Topic Expansion) dataset

5 Supervised Candidate Classification

We experimented with two complementary supervised classification methods: feature-based classification, which integrates diverse types of evidence from various sources, and a distantly-supervised neural network, which learns to discriminate between positive and negative pairs based on the contexts in which they co-occur.

5.1 Feature-Based Classification

We train a logistic regression classifier for each expansion type. The classifiers make use of novel sets of features designed for this task. Most features are shared by both classifiers, and a few additional features were developed specifically for each task. Below we give an overview of the features extracted for a given (DC,EC) pair. A more detailed and complete description is found in Appendix A.

Similarity & relatedness. The similarity and relatedness measures s_D, s_R, defined in Section 3.2.

Wikipedia. The following features take advantage of DC and EC both being Wikipedia titles, and make use of information found in their respective pages: (i) Number of Wikipedia categories shared by DC and EC; (ii) Count of occurrences of DC in EC categories or EC in DC categories

[6] https://idebate.org/debatabase

up to two category levels; (iii) count of shared Wikipedia outlinks of DC and EC.

WordNet. Whether DC is a hypernym, hyponym, synonym or co-hyponym of EC in WordNet (Miller, 1995) - four binary features.

Sentiment. Consistent expansions are expected to have the same sentiment polarity as the DC, whereas opposite polarities may indicate contrastive expansions (e.g., *Democracy* vs. *Dictatorship*). Similar to Iyyer et al. (2015), we train a linear SVM classifier on the sentiment lexicon of Hu and Liu (2004), using the word2vec word embeddings computed over the $\mathcal{C}_N$ corpus as the features and the word polarities as the labels. Word polarity can then be determined by the sign of the classifier's output score, and the sentiment strength by its magnitude. We take the product of the classifier's scores for DC and EC as a single sentiment feature.

Corpus statistics. Simple corpus-based features are derived from the number of co-occurrences of DC and EC in the same sentence in $\mathcal{C}_N$ or in the same query in $\mathcal{C}_Q$. These features are normalized to $[0, 1]$ by setting for each feature an upper threshold k on the count. Counts in the range of $[0, k]$ are linearly transformed to $[0, 1]$, and counts above k are set to 1. We also consider other corpus-based measures, such as pointwise mutual information (PMI) between DC and EC. For the consistent expansions classifier we also use as a feature the frequency ratio measure, defined in Section 3.2.

Other corpus-based features are based on pattern matching. For instance, we define a set of contrastive patterns, e.g, 'X *vs* Y' and 'X *instead of* Y'[7], and derive features such as the (normalized) count of (DC,EC) matches for these patterns, and the PMI of DC and EC in the subset of sentences/queries matching the patterns.

Overall, the feature count for the consistent expansions classifer is 15, and 22 for the contrastive expansions classifier.

5.2 Distantly Supervised Neural Network

The other classification approach we experimented with is based on distant supervision (Mintz et al., 2009). As before, we train two separate classifiers for consistent and for contrastive expansions, using their respective training sets. For each pair (DC,EC) from the training set, we retrieve from the $\mathcal{C}_N$ index up to 10,000 sentences that contain mentions of both DC and EC. The retrieved sentences are all labeled with the pair's label - positive or negative. These labels are noisy, since not every co-occurrence of DC and EC in a sentence is indicative of the relation between them. Our hope, however, is that the large number of training sentences collected this way would compensate for the noisy labels. The mentions of DC and EC in each sentence are replaced with generic symbols, `DC` and `EC`, to facilitate generalization over specific instances. We found that for consistent expansions, it is better to keep only the text between these two symbols, while for contrastive expansions, using the whole sentence works better. We balance the dataset to have an equal number of positive and negative training instances for each type.

The sentences collected for the whole training set are then used to train a neural network. Essentially, the network aims to determine whether a given sentence is a positive or a negative evidence for the existence of the target relation (consistent or contrastive expansion) between `DC` and `EC`. When applying the classifier to a new pair, we collect up to 500 sentences for that pair, and average the classifier's predictions for each sentence.

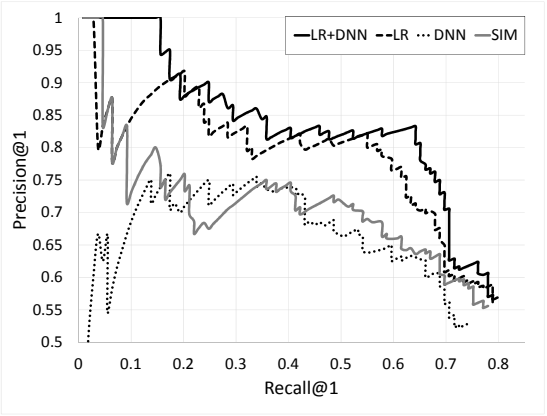
Neural network description. Our network is a bi-directional LSTM (Graves and Schmidhuber, 2005) with an additional attention layer (Yang et al., 2016). The models are all trained with a dropout of 0.85, using a single dropout across all timesteps as proposed by Gal and Ghahramani (2016). The cell size in the LSTM layers is 128, and the attention layer is of size 100. We use the Adam method as an optimizer (Kingma and Ba, 2015) with a learning rate of 0.001. Words are represented using the 300 dimensional GloVe embeddings learned on 840B Common Crawl tokens and are left untouched during training (Pennington et al., 2014).

6 Experiments

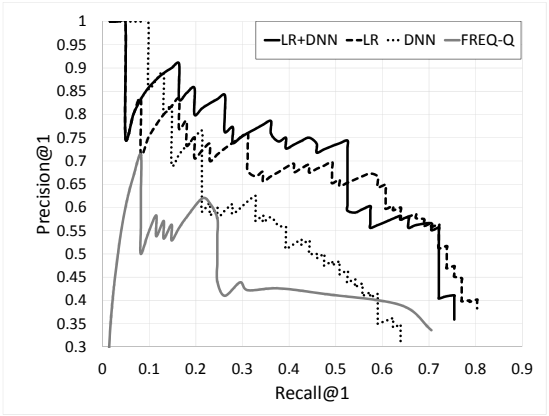
6.1 Experimental Setup

We assess the performance of our method on the following practical task: given a debate concept DC, find one good expansion concept EC for each expansion type. Recall that our dataset includes annotations for all the expansion candidates

[7] This set of patterns partially overlaps with the contrastive patterns described in Section 3.1.



(a) Consistent expansions



(b) Contrastive expansions

Figure 1: Comparison of candidate scoring methods

found for each DC by the candidate acquisition algorithm. Here we compare different methods for choosing one good expansion from these candidates. For each expansion type, we assume a scoring function $f(X, Y)$ over a pair of concepts, which predicts the likelihood of the target relation holding between X and Y. We further assume a threshold α representing the minimum score for a good expansion. Given a debate concept DC, we choose its highest scoring expansion, if its score exceeds the threshold. If no expansion was found, or all the expansion scores are below the threshold, we make no prediction. By modifying the threshold, we can explore the tradeoff between the number of predictions we make and their quality.

The following scoring functions were assessed:

Unsupervised baselines. (i) **SIM:** the distributional similarity measure s_D ; (ii) **REL:** the semantic relatedness measure s_R; (iii) **FREQ-N** co-occurrence frequency in $\mathcal{C}_N$; (iv) **FREQ-Q:** co-occurrence frequency in $\mathcal{C}_Q$.

Supervised Classification methods. (i) **LR:** the output score of the logistic regression classifier (Section 5.1); (ii) **DNN:** the output score of the distantly-supervised neural network (Section 5.2); (iii) **LR+DNN:** a simple combination of the LR and DNN classifiers, defined as the sum of their outputs.

Data and training. The 632 debate concepts in the dataset were split into a test set (230 concepts), a train set (295 concepts) and a development set (107 concepts). The train and development set for the DNN classifiers contain in total, for consistent expansions 736,305 sentences per class (positive/negative), and for contrastive expansions 252,298 sentences per class. The DNN classifiers were trained on the train portion of this dataset for 5 epochs, each resulting in a different model, and the best preforming model on the development set was chosen.

Due to the small number of instances in the development set, we did not use it to tune the LR classifier, but rather used both the train and the development sets to train the classifier. Together they contain 983/626 consistent/contrastive expansion candidates, respectively.

Performance measures. Let N be the total number of debate concepts in the test set, let C be the number of correct predictions, let P be the number of predictions made, and let R be the number of debate concepts in the test set for which good expansions exist. We define the following measures: (i) *Precision@1*= $\frac{C}{P}$; (ii) *Recall@1*= $\frac{C}{R}$; and (iii) *Coverage*= $\frac{P}{N}$.

6.2 Results

Figure 1 compares the above candidate scoring methods for both consistent (a) and contrastive (b) expansions. For each configuration, the Precision@1 vs Recall@1 graph is obtained by modifying the threshold α. Only the best-performing baseline for each expansion type is shown, for readability. Both the LR and the LR+DNN configurations outperform the strongest baseline by a large margin. This result illustrates the importance of supervised learning for this task.

For consistent expansions, LR+DNN is clearly the best-preforming configuration. For contrastive expansions, it outperforms the LR classi-

	Recall@1	
Precision@1	Consistent Expansions	Contrastive Expansions
0.9	0.248	0.164
0.8	0.651	0.262
0.7	0.706	0.525
0.6	0.780	0.574
0.5	0.798	0.721

Table 3: LR+DNN – Recall@1 for selected Precision@1 values.

fier in high-precision/low recall areas (Recall@1$<$ 0.53). For high-recall/low-precision areas, the LR classifier performs better.

As one may expect, the performance for consistent expansions is better than the performance for contrastive expansions, as the latter seems a more challenging task. Interestingly, for consistent expansions, SIM is the strongest baseline, whereas for contrastive expansions the best baseline is FREQ-Q. The performance of the DNN for consistent expansions is comparable to the best baseline, but for contrastive expansions it is much higher. Again, this may be attributed to difference in the difficulty of the two tasks, which requires, for contrastive expansions, more powerful methods.

We now take a closer look at the results for the LR+DNN configuration. To illustrate its performance, Table 3 includes sample data points for this configuration, for each expansion type.[8]

So far we used the Recall@1 measure to compare the coverage of different scoring methods with respect to the *given* set of candidates. Thus, the coverage of the candidate acquisition step was not taken into account in this assessment. In order to assess the end-to-end performance of our system, we next consider the tradeoff between Precision@1 and Coverage, as the latter measures the fraction of debate concepts for which we make a prediction out of *all* debate concepts in the test set.

The LR+DNN results for both expansion types are shown in Figure 2, and sample values are shown in Table 4. For example, by setting the threshold appropriately, we can find consistent expansions for 38.3% of the debate concepts with (at least) 80% precision. Precision and coverage for contrastive expansions are lower. For example, when requiring precision of 70%, we can make predictions for nearly 20% of the topics.

[8]For each given Precision@1 value p, we look for the maximal Recall@1 value such that its corresponding Precision@1 is at least p.

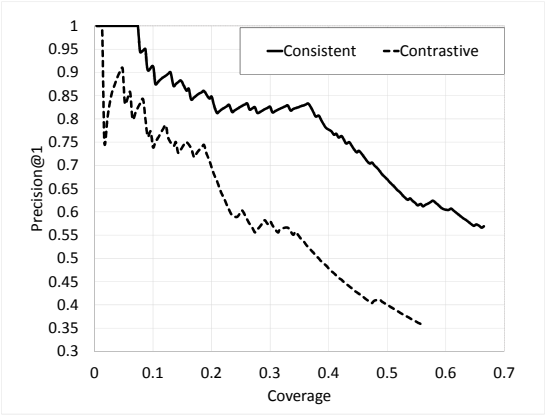


Figure 2: LR+DNN Precision@1 vs. Coverage

	Coverage	
Precision@1	Consistent Expansions	Contrastive Expansions
0.9	0.130	0.048
0.8	0.383	0.087
0.7	0.478	0.196
0.6	0.613	0.252
0.5	0.665	0.383

Table 4: LR+DNN – Coverage for selected Precision@1 values.

7 Related Work

There is a vast body of research work on identifying semantic relations between a pair of terms. Most studied relations include hyponyms/hypernyms, synonyms, antonyms, and meronyms. The main approaches applied to this task are summarized below.

Pattern-based methods. A fundamental type of evidence for detecting such relations is based on co-occurrence of the two terms in some text, typically in the same sentence. Pattern-based methods define lexico-syntactic contexts containing slots to be filled by instances of the target relation. Patterns can be defined over surface forms, or over syntactic representations such as paths in a dependency parse. Hearst (1992) introduced a pattern-based method for hyponym extraction, using a small set of manually-constructed textual patterns (for example "NP1 such as NP2"). Similar methods were used by Berland and Charniak (1999) for extracting meronyms, and by Lin et al. (2003) for identifying non-synonyms among semantically similar words.

Snow et al. (2005) developed a method to automatically learn new path-based patterns and used these patterns as features for hypernym classification; they later expanded this method for taxonomy construction (Snow et al., 2006). Schulte im Walde and Köper (2013) used automatically acquired word patterns to distinguish between antonyms, synonyms and hypernyms in German.

Distant supervision. Mintz et al. (2009) introduced the concept of distant supervision for relation extraction. The idea is to use an external knowledge base as a source of supervision instead of using labeled text (Riedel et al., 2010). The distant supervision paradigm assumes that any sentence that contains an entity pair of a known relation is likely to express the relation in the text. Since this assumption leads to noisy data and features, researchers have developed multi-instance approaches to deal with invalid sentences and wrong labels (Zeng et al., 2015; Riedel et al., 2010; Surdeanu et al., 2012). Another solution for the noisy data problem is using sentence-level attention model (Lin et al., 2016).

Distributional methods. Distributional methods aim to determine the relation between two terms by independently modeling the contexts in which each term occurs. Lin (1998) and later Weeds and Weir (2003) developed distributional similarity measures and showed that they can be used to predict hypernymy relations over WordNet terms.

Translation in a word embedding space may capture various syntactic and semantic relations between the words. This was demonstrated by Mikolov et al. (2013b) and Pennington et al. (2014) on the task of solving word analogies. Word embeddings were used for various relation extraction tasks, by taking their difference (Roller et al., 2014) or their concatenation (Baroni et al., 2012). Nguyen et al. (2016) used a distributional-based method for distinguishing antonyms and synonyms. Roller et al. (2018) compared the performance of Hearst patterns with distributional methods for hypernymy prediction and showed that co-occurrence measures of pairs extracted by Hearst patterns outperforms the distributional methods.

Neural approaches. Following recent advances in deep learning, many neural network architectures for relation classification have been proposed. Vu et al. (2016) combine recurrent and convolutional neural networks for relation classification. Shwartz et al. (2016) combine dependency path embeddings and distributional information for hypernym detection, and Nguyen et al. (2017) present a pattern-based neural network for distinguishing antonyms and synonyms.

Taxonomy induction from Wikipedia. Apart from relation extraction, taxonomy induction over Wikipedia concepts and categories is another line of research that is related to the current work. Examples are WikiTaxonomy (Ponzetto and Strube, 2007), YAGO (Hoffart et al., 2013), and the Wikipedia Bitaxonomy project (Flati et al., 2014). As described by Gupta et al. (2016), these works utilize information about Wikipedia concepts, the category network and the link structure.

The current work makes the following contributions with respect to previous relation extraction work. First and foremost, it introduces and studies a new relation extraction task - finding consistent and contrastive expansions for a given debate topic. To address this challenging task, we propose a hybrid architecture that combines diverse knowledge sources and techniques. Another contribution of this work is a novel set of patterns, filters and features designed specifically for this task.

Stance classification. Consistent and contrastive relations were previously discussed in the stance classification literature. Somasundaran et al. (2009) refer to these relations as *same/alternative*, and use them in conjunction with discourse relations to improve the prediction of opinion polarity. However, they do not attempt to identify these relations, but rather take them from a labeled dataset. Bar-Haim et al. (2017), as part of their work on claim stance classification, developed a classifier that aims to distinguish consistent from contrastive relations defined between the sentiment targets of a claim and the debate proposition. By contrast, our work addresses both candidate acquisition and classification, and most candidates are neither consistent nor contrastive expansions.

8 Conclusion

This work introduced a new task, debate topic expansion, along with a corresponding benchmark dataset, which we plan to make publicly available. We presented a working solution for this challeng-

ing task that achieved promising empirical results. The best results are obtained by combining diverse methods and techniques: pattern-based extraction, a novel set of filters and classification features, and a distantly-supervised neural network.

Debate topic expansion may be highly valuable for argumentation mining. For instance, topic-related argument mining has many potential use cases, such as helping individuals and organizations make better decisions, enhancing civic discourse by identifying arguments raised in the media, and promoting critical thinking among students. Debate topic expansion can enhance the coverage of existing argument mining methods by matching relevant arguments that do not mention the given topic explicitly. In addition, distinguishing consistent and contrastive expansions may improve argument stance classification. We plan to pursue these research directions in future work.

References

Ehud Aharoni, Anatoly Polnarov, Tamar Lavee, Daniel Hershcovich, Ran Levy, Ruty Rinott, Dan Gutfreund, and Noam Slonim. 2014. A benchmark dataset for automatic detection of claims and evidence in the context of controversial topics. In *Proceedings of the First Workshop on Argumentation Mining*, pages 64–68, Baltimore, Maryland. Association for Computational Linguistics.

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: Volume 1, Long Papers*, pages 251–261. Association for Computational Linguistics.

Marco Baroni, Raffaella Bernardi, Ngoc-Quynh Do, and Chung-chieh Shan. 2012. Entailment above the word level in distributional semantics. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 23–32, Avignon, France. Association for Computational Linguistics.

Matthew Berland and Eugene Charniak. 1999. Finding parts in very large corpora. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 57–64, College Park, Maryland, USA. Association for Computational Linguistics.

Liat Ein Dor, Alon Halfon, Yoav Kantor, Ran Levy, Yosi Mass, Ruty Rinott, Eyal Shnarch, and Noam Slonim. 2018. Semantic relatedness of wikipedia concepts - benchmark data and a working solution. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*. European Language Resource Association.

Tiziano Flati, Daniele Vannella, Tommaso Pasini, and Roberto Navigli. 2014. Two is bigger (and better) than one: the wikipedia bitaxonomy project. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 945–955, Baltimore, Maryland. Association for Computational Linguistics.

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 1027–1035, USA. Curran Associates Inc.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18(5-6):602–610.

Amit Gupta, Francesco Piccinno, Mikhail Kozhevnikov, Marius Pasca, and Daniele Pighin. 2016. Revisiting taxonomy induction over wikipedia. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2300–2309. The COLING 2016 Organizing Committee.

Iryna Gurevych, Eduard H. Hovy, Noam Slonim, and Benno Stein. 2016. Debating Technologies (Dagstuhl Seminar 15512). *Dagstuhl Reports*, 5(12):18–46.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th Conference on Computational Linguistics - Volume 2*, COLING '92, pages 539–545, Stroudsburg, PA, USA. Association for Computational Linguistics.

Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich, and Gerhard Weikum. 2013. Yago2: A spatially and temporally enhanced knowledge base from wikipedia. *Artif. Intell.*, 194:28–61.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 168–177, New York, NY, USA. ACM.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1681–1691, Beijing, China. Association for Computational Linguistics.

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

J. R. Landis and G. G. Koch. 1997. The measurements of observer agreement for categorical data. *Biometrics*, 33:159–174.

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

Ran Levy, Shai Gretz, Benjamin Sznajder, Shay Hummel, Ranit Aharonov, and Noam Slonim. 2017. Unsupervised corpus–wide claim detection. In *Proceedings of the 4th Workshop on Argument Mining*, pages 79–84. Association for Computational Linguistics.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning*, ICML '98, pages 296–304, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Dekang Lin, Shaojun Zhao, Lijuan Qin, and Ming Zhou. 2003. Identifying synonyms among distributionally similar words. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, IJCAI'03, pages 1492–1493, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2124–2133, Berlin, Germany. Association for Computational Linguistics.

Marco Lippi and Paolo Torroni. 2015. Context-independent claim detection for argument mining. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, pages 185–191. AAAI Press.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, NIPS'13, pages 3111–3119, USA. Curran Associates Inc.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia. Association for Computational Linguistics.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore. Association for Computational Linguistics.

Kim Anh Nguyen, Sabine Schulte im Walde, and Ngoc Thang Vu. 2016. Integrating distributional lexical contrast into word embeddings for antonym-synonym distinction. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 454–459, Berlin, Germany. Association for Computational Linguistics.

Kim Anh Nguyen, Sabine Schulte im Walde, and Ngoc Thang Vu. 2017. Distinguishing antonyms and synonyms in a pattern-based neural network. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 76–85, Valencia, Spain. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Simone Paolo Ponzetto and Michael Strube. 2007. Deriving a large scale taxonomy from wikipedia. In *Proceedings of the 22Nd National Conference on Artificial Intelligence - Volume 2*, AAAI'07, pages 1440–1445. AAAI Press.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Proceedings of the 2010 European Conference on Machine Learning and Knowledge Discovery in Databases: Part III*, ECML PKDD'10, pages 148–163, Berlin, Heidelberg. Springer-Verlag.

Stephen Roller, Katrin Erk, and Gemma Boleda. 2014. Inclusive yet selective: Supervised distributional hypernymy detection. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1025–1036, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Stephen Roller, Douwe Kiela, and Maximilian Nickel. 2018. Hearst patterns revisited: Automatic hypernym detection from large text corpora. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 358–363. Association for Computational Linguistics.

Sabine Schulte im Walde and Maximilian Köper. 2013. Pattern-based distinction of paradigmatic relations for german nouns, verbs, adjectives. In *Language Processing and Knowledge in the Web - 25th International Conference, GSCL 2013, Darmstadt, Germany, September 25-27, 2013. Proceedings*, pages 184–198.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2389–2398, Berlin, Germany. Association for Computational Linguistics.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2005. Learning syntactic patterns for automatic hypernym discovery. In Lawrence K. Saul, Yair Weiss, and Léon Bottou, editors, *Advances in Neural Information Processing Systems 17*, pages 1297–1304. MIT Press, Cambridge, MA.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2006. Semantic taxonomy induction from heterogenous evidence. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 801–808, Sydney, Australia. Association for Computational Linguistics.

Swapna Somasundaran, Galileo Namata, Janyce Wiebe, and Lise Getoor. 2009. Supervised and unsupervised methods in employing discourse relations for improving opinion polarity classification. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 170–179, Singapore. Association for Computational Linguistics.

Christian Stab, Johannes Daxenberger, Chris Stahlhut, Tristan Miller, Benjamin Schiller, Christopher Tauchmann, Steffen Eger, and Iryna Gurevych. 2018a. Argumentext: Searching for arguments in heterogeneous sources. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 21–25. Association for Computational Linguistics.

Christian Stab, Tristan Miller, Benjamin Schiller, Pranav Rai, and Iryna Gurevych. 2018b. Cross-topic argument mining from heterogeneous sources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3664–3674. Association for Computational Linguistics.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance multi-label learning for relation extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 455–465, Jeju Island, Korea. Association for Computational Linguistics.

Ngoc Thang Vu, Heike Adel, Pankaj Gupta, and Hinrich Schütze. 2016. Combining recurrent and convolutional neural networks for relation classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 534–539. Association for Computational Linguistics.

Henning Wachsmuth, Martin Potthast, Khalid Al Khatib, Yamen Ajjour, Jana Puschmann, Jiani Qu, Jonas Dorsch, Viorel Morari, Janek Bevendorff, and Benno Stein. 2017. Building an argument search engine for the web. In *Proceedings of the 4th Workshop on Argument Mining*, pages 49–59. Association for Computational Linguistics.

Julie Weeds and David Weir. 2003. A general framework for distributional similarity. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 81–88.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489. Association for Computational Linguistics.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762. Association for Computational Linguistics.

Appendices

A Features Used by the Logistic Regression Classifiers

Section A.1 lists all features used by either the consistent or contrastive classifiers (or both). ♠ marks features used by the consistent classifier; ♣ marks features used by the contrastive classifier. Some of the features use auxiliary definitions; those are listed in Section A.2. Some of the features are normalized as follows: let n be a normalization factor and f be the feature value. Then

$$f_n = \begin{cases} \frac{f}{n} & f \leq n \\ 1 & \text{otherwise} \end{cases}$$

In such cases, we mark the feature with $\mathcal{N}_n$.

A.1 Features

1. $\mathcal{C}_Q$**_TOT_COUNT**: count of DC, EC co-occurrences in $\mathcal{C}_Q$ queries. ♣$\mathcal{N}_{5000}$

2. $\mathcal{C}_Q$**_VS_COUNT**: count of DC, EC co-occurrences in $\mathcal{C}_Q$ queries matching a VS extraction pattern (see Section B.2). ♣$\mathcal{N}_{5000}$

3. $\mathcal{C}_Q$**_PMI**: *pointwise mutual information (PMI)* of DC, EC over all $\mathcal{C}_Q$ queries. ♣
$p(\text{DC}) = c_Q\text{\_DC\_COUNT}/c_Q\text{\_SIZE}$
$p(\text{EC}) = c_Q\text{\_EC\_COUNT}/c_Q\text{\_SIZE}$
$p(\text{DCEC}) = c_Q\text{\_TOT\_COUNT}/c_Q\text{\_SIZE}$

$$pmi(\text{DC}, \text{EC}) = \begin{cases} \log \frac{p(\text{DCEC})}{p(\text{DC}) \cdot p(\text{EC})} & p(\text{DC}) > 0 \\ & p(\text{EC}) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

4. $\mathcal{C}_Q$**_VS_PMI**: PMI of DC, EC over $\mathcal{C}_Q$ queries matching some VS extraction pattern. Same definition as Equation (1), but with the following quantities:
$p(\text{DC}) = c_Q\text{\_DC\_VS\_COUNT}/c_Q\text{\_VS\_SIZE}$
$p(\text{EC}) = c_Q\text{\_EC\_VS\_COUNT}/c_Q\text{\_VS\_SIZE}$
$p(\text{DCEC}) = c_Q\text{\_VS\_COUNT}/c_Q\text{\_VS\_SIZE}$
♠♣

5. **SIMILARITY_R**: semantic relatedness s_R between DC and EC (defined in Section 3.2). ♠♣

6. **SIMILARITY_D**: distributional similarity s_D between DC and EC (defined in Section 3.2). ♠♣

7. **EC_DC_SENTIMENT**: the product of DC concept sentiment and EC concept sentiment. ♠♣

8. **CATEGORIES_SHARED_COUNT**: count of Wikipedia categories shared by DC and EC. ♠

9. **CATEGORIES_CONTAINED_COUNT**: count of occurrences of DC in EC category names and EC in DC category names, up to two category levels; occurrences are counted after stemming both the category and the concept. ♠♣

10. **OUTLINKS_COUNT**: count of shared Wikipedia outlinks of DC and EC. ♠♣$\mathcal{N}_{30}$

11. **DC_HYPERNYM_EC**: a binary feature indicating if DC is a hypernym of EC in WordNet. ♠♣

12. **DC_HYPONYM_EC**: a binary feature indicating if DC is a hyponym of EC in WordNet. ♠♣

13. **DC_SYNONYM_EC**: a binary feature indicating if DC is a synonym of EC in WordNet. ♠♣

14. **DC_COHYPONYM_EC**: a binary feature indicating if DC is a co-hyponym of EC in WordNet. ♠♣

15. $\mathcal{C}_N$**_TOT_ALL**: count of DC, EC co-occurrence in $\mathcal{C}_N$ sentences, all surface forms, with distance of at most 10 tokens. ♠♣$\mathcal{N}_{5000}$

16. $\mathcal{C}_N$**_VS_ALL**: count of DC, EC co-occurrence in $\mathcal{C}_N$ sentences matching a VS classification pattern (see Section B.3.1), all surface forms. ♠♣$\mathcal{N}_{100}$

17. $\mathcal{C}_N$**_VS_EXACT_MATCH**: count of DC, EC co-occurrence in $\mathcal{C}_N$ sentences matching a VS classification pattern, exact surface forms. ♣$\mathcal{N}_{100}$

18. $\mathcal{C}_N$**_DEBATE_ALL**: count of DC, EC co-occurrence in $\mathcal{C}_N$ sentences matching a DEBATE classification pattern (see Section B.3.2), all surface forms. ♣$\mathcal{N}_{100}$

19. $\mathcal{C}_N$**_DEBATE_EXACT_MATCH**: count of DC, EC co-occurrence in $\mathcal{C}_N$ sentences matching a DEBATE classification pattern, exact surface forms. ♣$\mathcal{N}_{10}$

20. $\mathcal{C}_N$**_AND_ALL**: count of DC, EC co-occurrence in $\mathcal{C}_N$ sentences matching an AND classification pattern (see Section B.3.3), all surface forms. ♣$\mathcal{N}_{100}$

21. $\mathcal{C}_N$**_PATTERN_PROB_ALL**: probability of VS classification pattern, all surface forms. ♠♣

$$p(\text{DC}, \text{EC}) = \begin{cases} \log \frac{c_N\text{\_VS\_ALL}}{c_N\text{\_TOT\_ALL}} & c_N\text{\_VS\_ALL} > 0 \\ & c_N\text{\_TOT\_ALL} > 0 \\ -10 & \text{otherwise} \end{cases}$$

22. $\mathcal{C}_N$**_PATTERN_PROB_EXACT_MATCH**: Probability of VS classification pattern, exact match. ♣

23. $\mathcal{C}_N$**_VS_AND_RELATION**: A feature based on the ratio between the frequencies of the VS classification pattern and the AND classification pattern. ♣

$$p(\text{DC}, \text{EC}) = \begin{cases} \log \frac{\mathcal{C}_N\text{\_VS\_ALL}+1}{\mathcal{C}_N\text{\_TOT\_ALL}+1} & c_{\mathcal{C}_N\text{\_VS\_ALL}} > 0 \\ -10 & \text{otherwise} \end{cases}$$

24. $\mathcal{C}_N$**_FREQ_RATIO**: The frequency ratio measure, defined in Section 3.2. ♠

A.2 Auxiliary Definitions

1. $\mathcal{C}_Q$**_SIZE**: total count of $\mathcal{C}_Q$ queries.
2. $\mathcal{C}_Q$**_VS_SIZE**: total count of $\mathcal{C}_Q$ queries matching a VS pattern.
3. $\mathcal{C}_Q$**_DC_COUNT**: count of DC occurrences in $\mathcal{C}_Q$ queries.
4. $\mathcal{C}_Q$**_EC_COUNT**: count of EC occurrences in $\mathcal{C}_Q$ queries.
5. $\mathcal{C}_Q$**_DC_VS_COUNT**: count of DC occurrences in $\mathcal{C}_Q$ queries matching a VS pattern.
6. $\mathcal{C}_Q$**_EC_VS_COUNT**: count of EC occurrences in $\mathcal{C}_Q$ queries matching a VS pattern.
7. $\mathcal{C}_N$**_TOT_DC**: count of DC occurrences in $\mathcal{C}_N$ sentences, all surface forms.
8. $\mathcal{C}_N$**_TOT_EC**: count of EC occurrences in $\mathcal{C}_N$ sentences, all surface forms.

B Patterns

In the patterns listed in this section, (X,Y) stand for either (DC,EC) or (EC,DC), '*' matches any number of non-concept tokens, and [] indicates optional characters.

B.1 Patterns Used for Consistent Candidate Extraction

1. X is a Y
2. X is an Y
3. X is a kind of Y
4. X is a form of Y
5. X is an example of Y
6. X is a special case of Y
7. X or other Y
8. X or other types of Y
9. X or other kinds of Y
10. X or another type of Y
11. X and other Y
12. X and other types of Y
13. X and other kinds of Y
14. Y such as X
15. Y including X
16. Y e.g. X

B.2 VS Patterns Used for Contrastive Candidate Extraction

1. X * vs[.] * Y
2. X * versus * Y
3. X * preferable to * Y
4. X * instead of * Y
5. X * in contrast to * Y
6. X * better than * Y
7. X * healthier than * Y
8. X * safer than * Y
9. X * cleaner than * Y
10. difference[s] between * X * and * Y

B.3 Patterns Used for Candidate Classification

B.3.1 VS Patterns

1. X v[.] Y
2. X vs[.] Y
3. X versus Y
4. X instead of Y
5. X in contrast to Y
6. X [is|are] preferable to Y
7. X [is|are] better than Y
8. X [is|are] healthier than Y
9. X [is|are] safer than Y
10. X [is|are] cleaner than Y

B.3.2 DEBATE Patterns

1. X ⟨VS⟩ Y (debate|controversy); ⟨VS⟩ stands for any of the connectors listed in Section B.3.1.

2. X (and|or) Y (debate|controversy)

B.3.3 AND Patterns

1. both X (and|or) Y

2. including X (and|or) Y

3. such as X (and|or) Y

4. for example X (and|or) Y

5. for instance X (and|or) Y