Comparison of Three Similarity Measures for Selecting an Answer in a Retrieval-Based Chatbot

Olta Cakaj
Bremen, Germany
olta@uni-bremen.de

Vanja Sophie Cangalovic
Bremen, Germany
vanja@uni-bremen.de

Yvonne Jenniges
Bremen, Germany
yvo_jen@uni-bremen.de

ABSTRACT

A main challenge of retrieval-based conversational agents is the choice of a model to select an answer from a predefined set of responses. For this purpose, different methods are available. In the scope of the development of such an agent, three methods are compared in this work, namely cosine similarity over TF-IDF vectors, inner product over sentence embeddings and LDA. As dataset, discussion threads of online forums on the topic of climate change are used. Usually for a retrieval-based chatbot, a similarity measure is used to match the user's utterances to a question from the corpus. In a second step, the answers to that question are ranked by the similarity measure and the most suitable one is then returned. For the comparison of the three mentioned methods, top-level questions, i.e. questions opening a discussion thread, serve as an input for each method. The output, i.e. the returned response, is judged according to a metric that is based on the depth of the answer in the original conversation thread. As a result, cosine similarity and sentence embeddings performed comparable and achieved higher scores than LDA. This suggests that the former models are, in the chosen configuration, more suitable to serve as a text similarity measure on posts and to operate as answer selection algorithms in a retrieval-based chatbot.

UPDATED—April 27, 2020.

Author Keywords

conversational agents, retrieval-based, discussion platforms, text similarity measures

INTRODUCTION

Conversational agents can be classified into two main categories, according to their intended functionality. On the one hand, task oriented agents aim to support users in fulfilling a specific task, this setup is called a dialogue system. On the other hand, there are "chatbots", i.e. systems, which do not pursue a specific task, but are meant to have natural and meaningful discussions with people. [32]

For the implementation of a chatbot, two main architectures can be distinguished in general: generative and retrieval-based ones. Generative models assemble an answer given the user input and potentially some context knowledge, whereas retrieval-based models return an answer from the corpus that is ranked as best suited to the user input by some algorithm. Hence, besides the underlying dataset, the selection of an answer matching algorithm that retrieves a relevant answer from the corpus is a crucial step in the development of retrieval-based chatbots.

Measuring text similarity is an important part for a range of tasks, including text summarization, text classification, generation and answering of questions, as well as in information retrieval, for instance in retrieval-based chatbots. Gomaa et al. differentiate the existing measures into string-based, corpus-based and knowledge-based ones [16]. The string-based methods perform approximate string comparisons on the base of either terms or characters. Term-based measures include cosine and Jaccard similarity, while e.g. n-grams are considered to belong to the character-based subcategory. The second and the third group of text similarity measures, i.e. the corpus-based and the knowledge-based ones, are semantic measures. This means that they take into account what the context of the words is, the relation of the words to other words and how the words are used. The difference between corpus-based and knowledge-based methods is the data that drives their decision on similarity. While the corpus-based category relies on data from collected texts and/or speeches, knowledge-based measures rely on semantic networks, like WordNet. Examples for corpus-based methods are Latent Semantic Analysis (LSA) and Hyperspace Analogue Language (HAL). For instance, Leacock et al. implemented a knowledge-based measure [19]. Moreover, hybrid approaches combine techniques from multiple of the above mentioned categories. [16]

As a dataset, this work uses conversations from online discussion forums, since they contain an enormous and ever-growing amount of human communication about various topics, including climate change, which is the subject of focus for data collection. Since conversations in discussion forums are often arranged hierarchically, i.e. there can be answers to answers to a question, the corpus was flattened to question-answer pairs and each answer was assigned it's original hierarchy level for ease of evaluation.

In this paper, three answer matching algorithms for the retrieval process are compared: the traditional, string-based cosine similarity measure over TF-IDF vectors and the corpus-based inner product over sentence embeddings, as well as Latent Dirichlet Allocation (LDA) which is also corpus-based.

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

For the comparison, the models are evaluated by a metric which depends on the hierarchy level of the returned answer. In this way, statements can be made about the relevance of the models' answers without the need for manual annotation. Thus, this paper answers the **research question** (RQ):

> What are the differences in answer relevance of the three answer matching methods (cosine similarity, sentence embeddings, LDA) according to the employed evaluation metric?

The obtained results can help to choose a model in a retrieval-based chatbot.

RELATED WORK

For the purpose of this work, text similarity measures for a retrieval-based chatbot are evaluated. Hence, this section focuses on the implementations of this specific type of conversational agent architecture.

A general scheme in developing a retrieval-based model is to encode the corpus and then extract a relevant response by applying similarity measures. According to Li et al., cosine similarity is an extensively used method to match an appropriate response in information retrieval systems. [20]

Moreover, deep neural networks can be exploited for the task of encoding a natural language text into continuous-valued vectors. Most notably, word2vec and sentence embedding models have been proposed. [22, 4] The underlying networks are usually trained as language models. Word2vec's neural network of three layers, for example, can be trained via the Continuous Bag of Words and the Skip-Gram method. In these settings, the model's task is to predict either the current word given some surrounding words, or the context words, weighted by their distance, given a current word. [22] The state-of-the-art BERT model, in contrast, is primarily trained on the task of predicting the original value for a masked token, given a sequence of words taken from a large corpus. [11] The general idea being, that such networks are able to create task-agnostic representations for words or whole sentences in their first layer(s), which might then be applied and fine-tuned in other settings. [29] In [4] the authors introduce an improved baseline for sentence embedding through weighted average of word vectors using dimensionality reduction techniques such as PCA/SVD. Results of this paper show enhanced performance in calculating sentence similarity. One of the approaches proposed is a retrieval-based conversation system using deep learning to concatenate context utterances with the input message as redefined inquiries. [33] Additionally, another approach introduced is a subsequent matching scheme that matches answers with any utterance on multiple levels of granularity to retrieve relevant knowledge. [32] Furthermore, for multi-turn answer ranking in information retrieval systems, Yang et al. proposed a deep neural network learning structure which makes use of external knowledge. [35]

Minglai et al. offer an enhanced similarity measure using latent topic modelling and word co-occurrence analysis. To refrain from clustering inaccurate words that might have comparable topic probability but different topics nonetheless, linguistic correlation is evaluated through term co-occurrence given that the latter demonstrates an advanced text topic. [28]

Few studies compare different similarity measures for longer texts. A vast majority relates to the comparison of different vector space models regarding their ability to measure text similarity for short ones. Here, a short text means one of the length of a title, a text of medium length is e.g. an abstract and a long text depicts everything with more characters than that. In the following, related work on similarity measure comparisons for short texts is investigated. For instance, Lau and Baldwin compare doc2vec (document to vector) to two baseline methods, namely averaging word2vec (word to vector) as well as an n-gram model. [17] The former calculates the similarity of forum questions, whereas the latter aims to estimate the similarity between pairs of sentences. Based on this paper's findings, doc2vec performs noticeably better, especially when trained on large corpora. Another paper [23] compares three prominent models used to identify the semantic meaning of words, specifically a topic model (LSA), a neural network (word2vec), and GloVe (Global Vectors for Word Representation), which provides vector representations for words trained via an unsupervised algorithm based on the idea of higher co-occurrence probabilities for semantically similar words. Their results indicate that for the task of topic segmentation, word2vec constitutes a better word vector representation model than the other two employed methods.

Fewer research can be found on the comparison of similarity measures for longer texts. One example is the work of Dai et al., which aims to find the most appropriate similarity measure for whole paragraphs, by comparing doc2vec, respectively weighted word2vec representations, TF-IDF and LDA on two corpora, i.e. Wikipedia and arXiv. [8] The study concludes that doc2vec performs significantly better than the other models on the Wikipedia corpus. However, it scarcely outperforms TF-IDF on arXiv. In [27], an evaluation of various vectorisation techniques for encoding the semantics of natural language texts is conducted. Similarly to other papers mentioned above, the work examines TF-IDF, a topic model, in this case Latent Semantic Indexing (LSI), and paragraph vectors [18]. The latter are derived from neural networks, which have been trained to predict the words of a given document of variable length. The paper deduces that sophisticated text embedding techniques and extensions to Tf-IDF can barely outperform the original and simple TF-IDF method.

This work attempts to review existing methods used to measure similarity of texts that are on average of medium-length, compare them and provide guidance on which of the investigated measures yields better results. Following other research approaches, the compared models are a basic one, a topic model and a neural network. The three different response matching techniques employed are namely cosine similarity over TF-IDF vectors, LDA and inner product over sentence embeddings. Cosine similarity is a string-based, surface matching model [36], which is frequently employed as a baseline method. LDA is a corpus-based model and, according to the best of the authors' knowledge, less explored as a text similarity measure than LSA/LSI. The inner product over

sentence embeddings, another corpus-based method, is not represented in the considered comparative papers. Thus, this work integrates into the series of papers comparing text similarity measures. Contrary to the majority of presented papers, this work compares forum questions to other questions and answers from online discussion forums in the context of a retrieval-based chatbot.

METHODS

In order to answer the RQ, a retrieval-based chatbot was implemented with an interchangeable text similarity measure. On the basis of data from online discussion forums, the chatbot answers a user input by returning a response from the dataset that is selected by the similarity measure. The retrieved response is then evaluated using the evaluation metric presented in section 3.4.

Data Collection and Pre-processing

Online discussion platforms are considered an important medium for efficient and extensive participation [21] in discussions for a plethora of topics, among them climate-related issues. One of the main reasons for this is that they engage a high number of potentially diverse people in mutual reflection as well as exchanging viewpoints and knowledge. [10] Thus, by extracting data from online discussion forums, not only substantial amount of discussions and responses is available, but also people's authentic opinions on a particular issue.

Therefore, the data that serves as a base to this work is extracted from various discussion platforms, namely: Quora [25], TedTalk [30], Stack Exchange [13, 12, 14, 15], Climate Debate [9], Skeptical Science [26], and NASA [24]. Taking into consideration that the focus of this paper is climate change, climate related topics are extracted from the above-mentioned websites by performing web scraping, mainly using Selenium [1]. The scraped corpus, containing the nested question-answer threads, includes discussions in the timespan from 2010 - 2020.

The pre-processing steps needed for the examined similarity measures are the removal of stopwords, which is based on the idea of eliminating non-discriminative words in order to reduce the dimensionality of the feature spaces to be created, and stemming, which is common practice for the LDA and TF-IDF models.

The conversation are kept as they appear in the online forums. All data taken from the discussion platforms is then used as a question-answer pair, which is considered the main corpus. It consists of 153,540 question-answer pairs from 2,580 different threads. 93.26% of the threads have only direct comments. The others consist of up to seven answer levels (figure 1). On average, each thread has 14 answers resulting from a large number of threads with few answers and some outlier threads containing many answers with a maximum of 120 replies. Analyzing the length of the texts shows that questions are generally shorter than their answers having a median length of 67, compared to a median length of 851 for their replies.

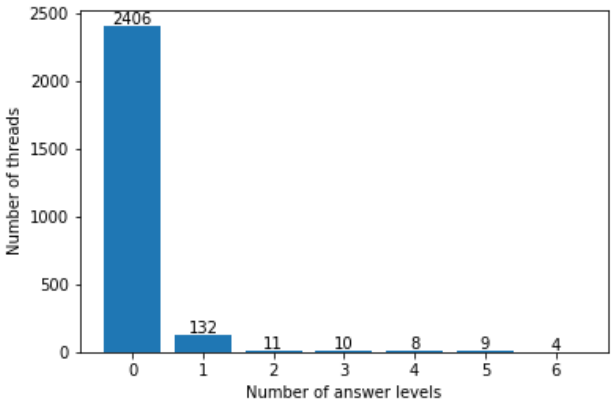

Figure 1. Depth of threads

Retrieval-Based Chatbot Architecture

Retrieval-based conversational agents are based on an existing corpus of predefined question-answer pairs, which can be created manually or based on already existing information. A retrieval-based chatbot applies a matching algorithm to rank the questions in the corpus and find the most similar one to the user utterance. The answer to the matched question is then returned to the user. [34]

For this work, a retrieval-based architecture is implemented. Since there are multiple cases in the scraped dataset, in which one question has more than one answer, the matching algorithm is first applied to all questions and afterwards to all answers from that question's thread. The key challenge, thereby, is to find a matching algorithm that returns a meaningful answer to the user. In the following, three different methods for this task are introduced.

Text Similarity Measures

The three methods employed in this work for calculating the similarity between two given natural language texts, i.e. TF-IDF vectorisation, sentence embeddings and LDA, are introduced and motivated in this chapter.

Cosine Similarity over TF-IDF Vectors

The idea of using cosine similarity calculated between TF-IDF vectorized texts follows the argumentation that utterances with similar words are also semantically similar.

Term Frequency - Inverse Document Frequency (TF-IDF) is a method to assign weights to terms in documents whereby two different weights are combined: the term frequency (TF) and the inverse document frequency (IDF). [2] Here, TF is the frequency of a term in a given document, which, in this work, corresponds to the frequency of a word in a given argument. The DF (document frequency) represents the fraction of the arguments containing this word, and IDF is a logarithmically scaled inverse DF. The latter is used to penalize common and semantically shallow words. [3] In order to then estimate the similarity between a given text and the corpus, pairwise cosine similarity calculates the angle between the respective vectors and returns a score between 0 and 1, respectively indicating

no similarity or identity. Using this traditional approach to approximate semantic similarity, while being extremely simple, has several disadvantages. It employs no knowledge of synonyms, thus texts about the same topic, but expressed with different, semantically related words would not be considered similar. Furthermore, it is a bag-of-words approach, meaning that all, potentially meaningful, syntagmatic structure is lost during the vectorization process.

Inner Product over Sentence Embeddings

Deep neural networks, trained on a variety of (self-)supervised tasks using large corpora of natural language utterances, provide another way to encode given sentences into continuous-valued vector representations. In order to reliably solve the various tasks, which can range from POS tagging, prediction of missing or subsequent words to causal reasoning, the networks need to learn an encoding in which words, or whole sequences of words, that are functionally similar, lie closer together. Thus, statistical and semantic information of the data is encoded. These latent representations can be retrieved as either word embeddings, a prominent example being word2vec (Mikolov et al., 2013), or sentence-level embeddings.

In this work, the Universal Sentence Encoder [7] is employed, which is based on the transformer architecture [31], and has been shown to deliver promising results in the domain of semantic textual similarity [6]. This pre-trained model is used with an embedding layer size of 512. The similarity between the resulting vectors is then calculated via their orthogonality, i.e. their pairwise inner product.

Topic Modeling - Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA) is a probabilistic method for topic modelling, i.e. for extracting topics from a number of documents. It is a form of unsupervised learning that observes documents as bags of words and characterizes a topic by a distribution over words. The assumption that each document consists of multiple topics in different proportions results in a topic distribution for each document. [5]

Compared to cosine similarity to calculate the similarity between documents, LDA offers the advantage that is also takes semantic factors into account such that not only the same words are matched but also words that belong to the same semantic area/topic. [28]

Evaluation Metric

The conversational agent needs to retrieve the most appropriate answer to a user input, given a large corpus of human conversations. In order to evaluate the correctness/accuracy of this task, a quantitative evaluation metric is needed. Since manually-annotated data would be both time- and labour-intensive, this work employs an unsupervised method using the corpus at hand. This existing corpus of natural language discussions contains answers to various questions, that have - evidently - been classified as appropriate and relevant by some human, i.e. the answer's author. This fact is utilized by evaluating the three aforementioned similarity metrics on these conversations. Therefore, the correctness of the system's output is determined by its belonging to the original input's thread, which is determined by a numerical score (Eqn. 1). Retrieving one of the direct answers is interpreted as a perfect score of 1, while answers from different threads as a failed 0, and answers further down the conversation hierarchy receive a discounted score. The discount depends on the hierarchy level of the answer, i.e. a direct answer to a question is assigned hierarchy level 0, a comment to a direct answer is on level 1 and so on. Furthermore, the level of the answer is divided by the overall maximum number of levels plus a constant number, whose purpose is to prevent a selected answer from the last hierarchy level of a thread to receive a score of 0.

$$score = 1 - \frac{hierarchy\_level}{max\_hierarchy\_level + const} \in [0,1] \qquad (1)$$

Evaluation Modes

The evaluation was conducted using the top-level questions as input. Thus, the outcome of the three similarity measures can be compared against the ideal answer (a direct comment to the question). In order to analyse the behaviour of the models, four different evaluation modes are applied which are abbreviated with question_binary, answer_default, answer_all and answer_correct_question. They are explained in the following. Since the first step of the answer retrieval is the matching of the user input to the most similar question from the corpus, the evaluation mode question_binary checks if the model can find the most similar question to the input out of all top-level questions in the corpus, which would be the exact same question. Thus, the question matching step is tested. The option answer_default orients itself on how a retrieval-based chatbot usually retrieves an answer: First, the most similar question is chosen from the top-level questions. Out of the answers to this selected questions, the most suitable one according to the model is returned. The third option is answer_all, which retrieves the most similar answer out of all the answers in the corpus. This can give insights about the relevance of the question matching step. Lastly, answer_correct_question retrieves the most similar answer from the thread of the original question to evaluate the output for the case that the question matching was successful.

Hyperparameters

For LDA, an important parameter that has to be manually chosen is the number of topics n. There is no established method to perform this task, but several are proposed in the scientific community. Here, a subjective evaluation was performed besides the calculation of perplexity. [37] These methods are employed for $n = 10, 20, 40, 50$ and $n = 100$ topics to cover a broad range. The manual check yielded similar results for all n: 20 - 30 % of the topics were not interpretable or irrelevant. The perplexity score, which describes the ability of a statistical model to depict a dataset [37], was more significant. Generally, a lower perplexity indicates a better generalization ability of the model. For the tested number of topics, the minimum perplexity was reached for $n = 20$ topics. Since the manual evaluation of the topics did not show any significant difference between the models, the perplexity score served as a means for the topic number selection. Hence $n = 20$ was selected.

RESULTS

The above mentioned evaluation was conducted for all three similarity measures for every evaluation mode, i.e. ques-

tion_binary, answer_all, answer_correct_question and answer_default. The results can be seen in Tab. 1, Tab. 2, Tab. 3 and Tab. 4. For each similarity measure, the tables depict how many of the 2580 input questions resulted in an output with the same score. For example, Tab. 2 (evaluation mode a_all) shows that cosine similarity returned in 2146 out of 2580 cases an answers that got a score of 0. The mode q_binary is limited to a score of either 0 or 1 because it states if the answer was in the original question thread ($score = 1$) or not ($score = 0$). The other three modes, answer_all, answer_correct_question and answer_default, can produce eight different scores according to Eqn. 1 since the present corpus exhibits a maximum of seven answer levels in addition to the score of 0 (representing that the returned answer was not located in the original question thread).

Cosine Similarity over TF-IDF Vectors

Calculating the cosine similarity between the same two TF-IDF vectorized texts results, by definition, in a score of 1. Thus, the question_binary evaluation task is fulfilled with an average score of 1 (Tab. 1). Due to the accurate results of the question_binary task, all of the matched answers in mode answer_default and answer_correct_question are correctly located in the respective discussion threads, leading to scores larger than 0 (Tab. 4 and Tab. 3). Thus, these two modes resulted in similar scores and are considered together here. The scores average a value of 0.99 over the 2580 input questions, and in 92% of the cases, first-level answers were selected. The answer_all task resulted in lower scores, the average being 0.168, having matched into the original thread in 16.8% of the cases and having selected a first-level answer in 15% (Tab. 2).

Inner Product over Sentence Embeddings

In line with the results reported for cosine similarity over TF-IDF vectorised texts, calculating the inner product over the same two embedded texts achieves perfect results, the average score of question_binary being 1 which is shown in Tab. 1. In consequence, the answer_default and answer_correct_question tasks resulted in similar scores, averaging a value of 0.998 over the 2580 input questions. In 95% of the cases, first-level answers were selected (Tab. 4 and Tab. 3). With an average score of 0.132, this approach resulted in a poorer performance on the more difficult answer_all task. In 13.3% of the cases, the measure selected an answer from the original thread (Tab. 2).

LDA

Overall, LDA achieved very low scores. This is already visible when evaluating the mode question_binary (Tab. 1): Apart from one exception, the majority of the matched questions did not equal the original thread question and therefore received a score of 0. Since the question is not matched correctly, the answer is picked from another question thread. This entails poor values for the evaluation option answer_default (Tab. 4) because the question selection is the first part of the latter evaluation method. For the option answer_all, LDA performed similar: In most cases, namely 99.996%, the retrieved answer was not included in the thread of the input question, and thus received a score of 0 (Tab. 2). Lastly, Tab. 3 shows, given the correct question, which level the retrieved answer has. Most answers are on the first level, i.e. direct answers to the top-level question.

score	Cosine Similarity	Sentence Embeddings	LDA
0.0	0	0	2579
1.0	2580	2580	1

Table 1. Scores of the similarity measures for question_binary

Analysis of Results

According to the employed evaluation metric, sentence embeddings and cosine similarity performed similar and outperformed LDA. Since LDA already fails to match the correct question, it is not able to retrieve an answer that would be labeled as suitable by the metric. Thus, it returns lower scores than the other two methods, excluding the case of answer_correct_question. In the latter mode, LDA achieved comparable results to the other two methods. However, it matched significantly more second level answers, 393 in total, compared to 130 (cosine similarity) and 99 (sentence embeddings). Therefore, the evaluation metric penalizes LDA, indicating that its answer matching capability is inferior to the abilities of cosine similarity and sentence embeddings, which return more direct, i.e. first-level, answers. The high number of first level answers for all three cases can be explained by the fact that the corpus mainly consists of threads which have first level answers only. Cosine similarity and sentence embeddings achieved very similar scores, since they always find the correct question. However, these two models rely on the question matching step for finding a relevant answer, which is inferable from Tab. 2: Without this step, i.e. when selecting the answer directly from all possible responses, the methods only choose direct answers to the input question in a small fraction of the cases. In this mode, cosine similarity performed slightly better than sentence embeddings choosing 91 times more often an answer that was in the original question thread. In contrast, sentence embeddings achieved marginally better scores for answer_correct_question with an average score of 0.9935 compared to an average score of 0.9913 (cosine similarity).

Ultimately, this comparison of the three employed text similarity measures answers the RQ: The difference between the models according to the introduced evaluation metric manifests in the computed scores. While cosine similarity and sentence embeddings achieve an average score of 1 when matching the question, LDA reaches an average score of 0. The answer matching on the base of the correct question, resulted in average scores of 0.9935 (sentence embeddings), 0.9913 (cosine similarity) and 0.9825 (LDA). Thus, cosine similarity and sentence embeddings yield comparable results which are better than those of LDA.

DISCUSSION

In line with Shahmirzadi et al. and Dai et al., the results of this paper indicate cosine similarity over TF-IDF vectors to be a good choice for measuring the similarity of documents [27, 8]. According to the former work, performance and cost of this model seem to justify its usage [27].

score	Cosine Similarity	Sentence Embeddings	LDA
0.0	2146	2237	2579
0.45	0	0	0
0.55	0	0	0
0.64	1	2	0
0.73	0	0	0
0.82	0	0	0
0.91	44	24	0
1.0	389	317	1

Table 2. Scores of the similarity measures for answer_all

score	Cosine Similarity	Sentence Embeddings	LDA
0.0	0	0	0
0.45	2	2	2
0.55	4	4	4
0.64	5	4	5
0.73	10	6	7
0.82	11	11	18
0.91	130	99	393
1.0	2418	2454	2151

Table 3. Scores of the similarity measures for answer_correct_question

score	Cosine Similarity	Sentence Embeddings	LDA
0.0	0	0	2579
0.45	2	2	0
0.55	4	4	0
0.64	5	4	0
0.73	10	6	0
0.82	11	11	0
0.91	130	99	0
1.0	2418	2454	1

Table 4. Scores of the similarity measures for answer_default

Moreover, Shahmirzadi et al. state that due to a possibly extensive parameter tuning, more complex models, like LSI and paragraph vectors, should only be employed on compact texts in which the similarity measurement is rather rough [27]. In our case of answer-retrieval on medium-length posts from online discussion forums, coarser matching could be beneficial to return semantically more relevant answers. However, for texts with a length exceeding that of a title, Shahmirzadi et al. found that their tuned LSI model performed worse than cosine similarity on Tf-IDF [27]. Hence, for online discussions, where posts are usually longer than a title, LSI would not be recommended. Though the present work uses LDA instead of LSI, it also comes to the conclusion that LDA is outperformed by the paragraph vectors and TF-IDF, substantiating the findings of Dai et al. [8]. However, it should be considered that different parameters for LDA might lead to different results. The combination of a topic model with other methods could as well improve the value of LDA as a text similarity measure, as Minglai et al. showed by combining LDA with word co-occurence. [28]

The neural network-based paragraph vectors employed by Shahmirzadi et al. performed, similar to their LSI model matching very short texts, better than cosine similarity over TF-IDF. However, this result was achieved only after extensive parameter tuning. [27] In contrast, the Universal Sentence Encoder employed in this work generally resulted in scores very similar to those of cosine similarity over TF-IDF, though it could not outperform this method. This difference might be explained by the varying underlying architectures of the models. While the paragraph vectors are explicitly trained to predict words of a given document, the Universal Sentence Encoder is trained via multi-task learning, creating the sentence embeddings along the way. Another explanation for the different results might be the different corpora, which possibly entail different linguistic challenges. Overall, the present findings agree with other related work (e.g. [23] and [8]) in suggesting that neural networks appear to provide promising ways to calculate text similarities.

Limitations

A major limitation of this work is the lack of the number of both discussions in general and deep discussion threads in particular, which severely restricts the relevance of the present evaluation results. In order to achieve more reliable results, a larger corpus would be needed.

Another fundamental limitation concerns the employed evaluation metric itself. Intuitively, the assumption of all answers in a given discussion thread treating the same topic and being relevant to the first question might appear far-fetched to online forum-experienced people. The tendency of human discussions to diverge from the initial topic is reflected in the discounting of the metric. However, it is realistic to assume that there are cases in which answers from different threads might actually be more relevant to a given question than comments from the original thread. This observation results in the possibility of having a high number of false negatives, which could only be detected by manual investigation of the results.

CONCLUSION

This work presents three text similarity measures for returning relevant, i.e. semantically related, answers in a retrieval-based conversational agent: cosine similarity over Tf-IDF, sentence embeddings and LDA. The models are (trained and) tested using question-answer pairs from online discussion forums, focusing on the topic of climate change. Moreover, a metric is introduced to automatically evaluate the three methods. The metric depends both on the hierarchy level of the answer, as well as the maximum number of answer levels in the present corpus. Although cosine similarity over TF-IDF vectors is a simple method, discarding much information, which is deemed crucial for understanding a text's precise semantics, it performs similarly to the inner product over sentence embeddings, whose sophisticated underlying neural network has potentially learnt layers of semantic information associated with natural language constructions. Furthermore, LDA, despite being a promising measure since it contains, contrary to Tf-IDF vectors, a semantic differentiation, performed very poorly.

Future Work

In order to improve the quality of the returned answers, several steps can be taken. One possibility is to improve LDA, e.g. by applying other measures for finding an appropriate number of topics. This could be useful because there is no established method to determine this parameter. A different measure could lead to a different number of topics and therefore might change the results for the presented evaluation.

Additionally, along the lines of ensemble methods in machine learning architectures, a combination of the presented similarity measures might be able to yield better results. For example, calculating LDA and cosine similarity would result in the texts being matched on both a semantic and a word-based statistical level.

REFERENCES

[1] 2020. SeleniumHQ Browser Automation. (2020). https://www.selenium.dev/ Accessed: 2020-04-16.

[2] Akiko Aizawa. 2003. An Information-Theoretic Perspective of Tf—Idf Measures. *Inf. Process. Manage.* 39, 1 (Jan. 2003), 45–65. DOI: http://dx.doi.org/10.1016/S0306-4573(02)00021-3

[3] M. Alodadi and V. P. Janeja. 2015. Similarity in Patient Support Forums Using TF-IDF and Cosine Similarity Metrics. In *2015 International Conference on Healthcare Informatics*. 521–522.

[4] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2019. A simple but tough-to-beat baseline for sentence embeddings.

[5] David M. Blei. 2012. Probabilistic Topic Models. *Commun. ACM* 55, 4 (Apr 2012), 77–84.

[6] Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, Vancouver, Canada, 1–14. DOI: http://dx.doi.org/10.18653/v1/S17-2001

[7] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal Sentence Encoder. *arXiv:1803.11175 [cs]* (April 2018). arXiv: 1803.11175.

[8] Andrew M. Dai, Christopher Olah, and Quoc V. Le. 2015. Document Embedding with Paragraph Vectors. *CoRR* abs/1507.07998 (2015). http://arxiv.org/abs/1507.07998

[9] Climate Debate. 2020. Climate debate in general. (2020). https://www.climate-debate.com/forum/climate-debate-in-general-f6.php Accessed: 2020-01-08.

[10] Liping Deng, Yang-Hsueh Chen, and Sandy C. Li. 2017. Supporting cross-cultural online discussion with formal and informal platforms: a case between Hong Kong and Taiwan. *Research and Practice in Technology Enhanced Learning* 12, 1 (2017), 5. DOI: http://dx.doi.org/10.1186/s41039-017-0050-z

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. (2018).

[12] Stack Exchange. 2020a. Questions tagged [climate]. (2020). https://earthscience.stackexchange.com/questions/tagged/climate Accesed: 2020-01-20.

[13] Stack Exchange. 2020b. Questions tagged [climate-change]. (2020). https://earthscience.stackexchange.com/questions/tagged/climate-change Accessed: 2020-01-20.

[14] Stack Exchange. 2020c. Questions tagged [climate-models]. (2020). https://earthscience.stackexchange.com/questions/tagged/climate-models Accesed: 2020-01-20.

[15] Stack Exchange. 2020d. Questions tagged [climatology]. (2020). https://earthscience.stackexchange.com/questions/tagged/climatology

[16] Wael H.Gomaa and Aly A. Fahmy. 2013. A Survey of Text Similarity Approaches. *International Journal of Computer Applications* 68, 13 (Apr 18, 2013), 13–18.

[17] Jey Lau and Timothy Baldwin. 2016. An Empirical Evaluation of doc2vec with Practical Insights into Document Embedding Generation. 78–86. DOI: http://dx.doi.org/10.18653/v1/W16-1609

[18] Quoc V. Le and Tomas Mikolov. 2014. Distributed Representations of Sentences and Documents. (2014).

[19] Claudia Leacock and Martin Chodorow. 1998. *Combining Local Context and WordNet Similarity for Word Sense Identification*. Vol. 49. 265–.

[20] Baoli Li and Liping Han. 2013. Distance Weighted Cosine Similarity Measure for Text Classification. In *Intelligent Data Engineering and Automated Learning – IDEAL 2013*. Springer Berlin Heidelberg, Berlin, Heidelberg, 611–618.

[21] Edith Manosevitch, Nili Steinfeld, and Azi Lev-On. 2014. Promoting online deliberation quality: cognitive cues matter. *Information, Communication & Society* 17, 10 (2014), 1177–1195. DOI: http://dx.doi.org/10.1080/1369118X.2014.899610

[22] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. (2013).

[23] Marwa Naili, Anja Habacha, and Henda Ben Ghezala. 2017. Comparative study of word embedding methods in topic segmentation. *Procedia Computer Science* 112 (12 2017), 340–349. DOI: http://dx.doi.org/10.1016/j.procs.2017.08.009

[24] NASA. 2020. Frequently Asked Questions. (2020). https://climate.nasa.gov/faq/ Accessed: 2020-01-25.

[25] Quora. 2020. Results for climate change. (2020). https://www.quora.com/search?q=climate+change Accessed: 2020-01-31.

[26] Sceptical Science. 2019. Global Warming Climate Change Myths. (2019). https://skepticalscience.com/argument.php Accessed: 2019-12-15.

[27] O. Shahmirzadi, A. Lugowski, and K. Younge. 2019. Text Similarity in Vector Space Models: A Comparative Study. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*. 659–666.

[28] Minglai Shao and Liangxi Qin. 2014. Text Similarity Computing Based on LDA Topic Model and Word Co-occurrence. In *2nd International Conference on Software Engineering, Knowledge Engineering and Information Engineering (SEKEIE 2014)*. Atlantis Press.

[29] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to Fine-Tune BERT for Text Classification? (2019).

[30] TedTalks. 2019. Talks. (2019). https://www.ted.com/search?cat=videos&q=climate Accessed: 2019-12-17.

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. (06 2017).

[32] Yu Wu, Wei Wu, Chen Xing, Ming Zhou, and Zhoujun Li. 2016. Sequential Matching Network: A New Architecture for Multi-turn Response Selection in Retrieval-based Chatbots. (2016).

[33] Rui Yan, Yiping Song, and Hua Wu. 2016b. Learning to Respond with Deep Neural Networks for Retrieval-Based Human-Computer Conversation System. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '16)*. Association for Computing Machinery, New York, NY, USA, 55–64. DOI:http://dx.doi.org/10.1145/2911451.2911542

[34] Zhao Yan, Nan Duan, Junwei Bao, Peng Chen, and Ming Zhou. 2016a. DocChat: An Information Retrieval Approach for Chatbot Engines Using Unstructured Documents. In *54th Annual Meeting of the Association for Computational Linguistics*. Berlin, Germany, 516–525.

[35] Liu Yang, Minghui Qiu, Chen Qu, Jiafeng Guo, Yongfeng Zhang, W. Bruce Croft, Jun Huang, and Haiqing Chen. 2018. Response Ranking with Deep Matching Networks and External Knowledge in Information-Seeking Conversation Systems. In *The 41st International ACM SIGIR Conference on Research Development in Information Retrieval (SIGIR '18)*. Association for Computing Machinery, New York, NY, USA, 245–254. DOI: http://dx.doi.org/10.1145/3209978.3210011

[36] Wen-Tau Yih and Christopher Meek. 2007. Improving similarity measures for short segments of text. In *AAAI*, Vol. 7. 1489–1494.

[37] Weizhong Zhao, James J. Chen, Roger Perkins, Zhichao Liu, Weigong Ge, Yijun Ding, and Wen Zou. 2015. A heuristic approach to determine an appropriate number of topics in topic modeling. In *12th Annual MCBIOS Conference*, Vol. 16 Suppl 13. https://www.ncbi.nlm.nih.gov/pubmed/26424364