Phrase to Phrase Similarity Scoring with Semantic Matching

Beyza Özen
1 July 2022

Introduction

Before being granted, patents go through a rigorous vetting process. With the accumulating number of patents and the number of patent applications reaching a new record each year, it is inevitable to use natural language processing (NLP) to avoid data redundancies by connecting the dots between millions of patent documents. Generation of summaries, structured information extraction, text mining, clustering and categorization, relevancy scoring, and language translation are some applications of NLP to global patent databases [1,2,3]. The given tasks can be achieved by models that do not require semantic processing, attempting to distill the meaning of words and phrases [4]. There are many studies on summarizing patents and extracting relevant information. The problem that occurs in these studies is the inability to control the semantic matching of the extracted phrases within a given context. The purpose of this study is to fine-tune a pre-trained model to match given phrases in order to extract contextual information using the U.S. Patent and Trademark Office (USPTO) open dataset and present a scoring of the relevance of the phrases parallel to the specified context.

Since the semantics can be learned within the given context, in previous studies, the texts that only cover the given area of interest were used to train the model. However, when we proceed with this method, the system gives an error when it encounters a term outside the learned area or makes an incorrect evaluation by sticking to the learned context. Even though it is possible to introduce texts for every subject to model to increase the area of understanding, it is inefficient. With this kind of consideration, we also need to specify the topics that are crossing

and possible new topics to stay up to date. In short, using a model that can understand semantics as well as related topics is a logical way to proceed considering the many pre-trained efficient system that exists currently.

Detecting phrase similarity requires understanding the semantics of inputs since different words and/or phrases can be used to describe the same/similar concepts or they can be paraphrases. For this reason, the used models should capture fine-grained word-level information for semantic comparisons instead of large-scale event-based systems. This study will focus on semantic matching scoring using the phrases extracted from the massive patent texts by the U.S. Patent and Trademark Office (USPTO).

With this assignment, it will be shown that the meanings of words or phrases can be learned with pre-trained systems by knowing the subject in general, without the need for a large text explanation.

Helping people who evaluate patents with this method is the reason for starting this study. However, the biggest contribution of this study is undoubted to prove the existence of a model that has learned the meanings of phrases in a contextual explanation that consists of only a few words, without the need for the whole text. The fact that the model can correctly evaluate the relationship between two phrases in the presence of a context proves that this model has learned the phrases as a whole with the words' real meanings and the side meanings. Thus, without spending resources to analyze the entire text; we can make sense of the terms in the text using only the general topic.

Related Works

Mikolov et al. [6] introduce a continuous skip-gram model that captures word-vector representations that can capture semantic word relationships. Therefore, they show that without

a deep neural network, the word vectors can be somewhat meaningfully combined using simple vector addition. Later, a similar methodology introduced in this paper was used for computing continuous bag-of-words and skip-gram architectures for vector representations of words from very large data sets to create the Word2vec algorithm [12] and GloVe [13]. Both models are performing as promised in the source reference [6]. Due to their ability to decrease high dimensional data into vectors, they require much lower computational resources, and they can easily handle large amounts of data efficiently and capture indirect relations between terms. However, they are context-independent and mostly trained by shallow models. The models lack of sufficiently represented the subjects related to the terminology on relations among the semantic units required for patent data.

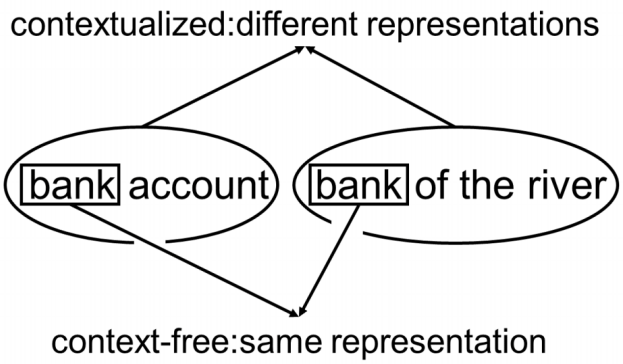

Figure 2: *For the above example, the context of the word "bank" is the same for context-free algorithms like Word2vec and GloVe. [15]*

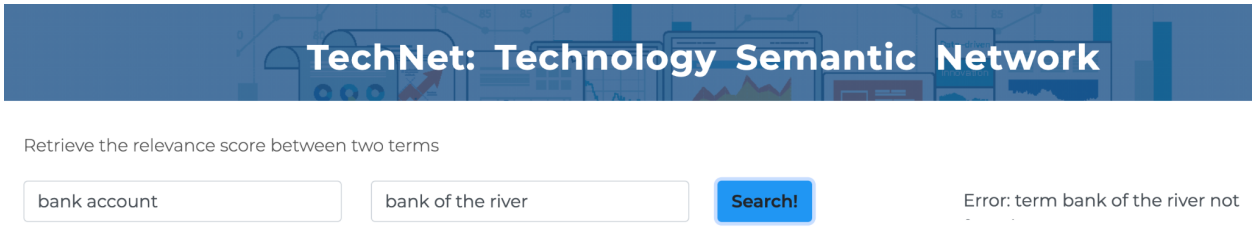

Figure 3: *For the example in figure 2, Technet does not have the vector for the second term since it is not related to the topics that the algorithm created. [http://www.tech-net.org/]*

Sarica et al. [14] focused on the context-based requirements of Word2vec and GloVe and create a new relational vector space focused on technology-related data. In order to create content space, they used an already classified patent text database and extract terms from massive technical patent texts. From the standpoint of technology and engineering design, retrieving terms and their pairwise relevance is the advantage of this work. Even though it outperformed the previous works on the specific domain, for the recent study it is not enough to cover all patent domains.

Schomacker et al. [15] compiled a review on the language representation models that outperform human performance in natural language processing tasks. Figure 4 summarises the models considered by the article in chronological order. The article also summarises key concepts for natural language models with attributes to related models. To broaden the scope of contextualized word embedding, the article suggests bidirectional representations. From figure 4, it is easily observed that after the creation of BERT and RoBERTa the latest form is the DeBERTa (2021). It is short for decoding-encoding BERT with disentangled attention. It bulbs on BERT and RoBERTa and improves their result by using a distance-based approach and a combination of absolute and relative positions approaches.

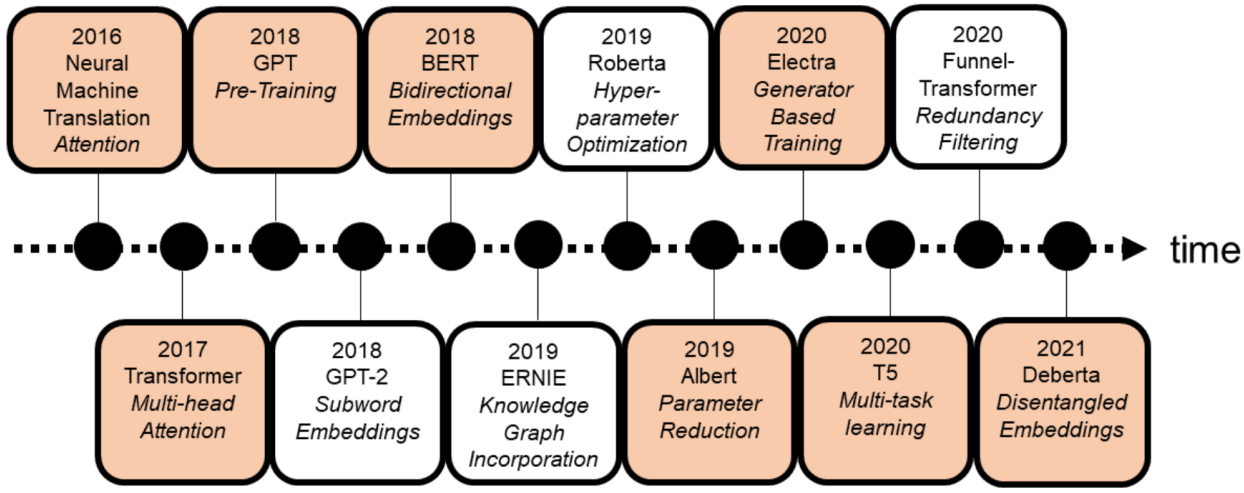

Figure 4: *The considered milestone models by Schomacker et al. in chronological order.*

Materials and Methods

In this study, the "Google Patent Phrase Similarity Dataset" is used for fine-tuning and performance measuring of the pre-trained DeBERTa model. The dataset was created by The U.S. Patent and Trademark Office (USPTO) to advance research on matters relevant to intellectual property. The collection was scanned by Google into Google Patents Public Datasets to create a collection of publicly accessible, connected database tables for empirical analysis of the international patent system.

Patent data is invaluable for studying historical and present-day innovation. Among 90 million published patents from 17 countries, it is very challenging to examine previous studies for newly developed patents or to find old patents related to the candidate patent that are evaluated to protect the intellectual properties after the patent application. The patent similarity dataset provides phrases level similarity measures for granted patents.

The provided dataset consists of word tuples and their corresponding average similarity scores evaluated by human participants in non-technical contexts. The dataset contains 5 columns namely; ID, anchor, target, context, and score. Anchor and target are the phrases that are compared under the context given in the format of Cooperative Patent Classification (CPC) [11]. The dataset contains more than 36k labeled entries with a given relevance score from 0 to 1 with increments of 0.25 where 1 means absolute match and zero means no relevance at all. The phrases can be considered as the keywords that one types on the search bar to get relative patent documents that contain similar phrases as keywords. In the context of the similarity dataset, the former phrase mentioned can be called anchor and the latter one is called target. The relevance that lists the patents is a score.

As it is known, the USPTO has long been granting U.S. patents and registered trademarks. During this time, patent authorities prepared similarity scores for phrases that were

matching in related subject fields. The score relationship between the phrases in the dataset provided by the USPTO has been provided by compiling this historical experience.

	id	anchor	target	context	score
0	37d61fd2272659b1	abatement	abatement of pollution	A47	0.50
1	7b9652b17b68b7a4	abatement	act of abating	A47	0.75
2	36d72442aefd8232	abatement	active catalyst	A47	0.25
3	5296b0c19e1ce60e	abatement	eliminating process	A47	0.50
4	54c1e3b9184cb5b6	abatement	forest region	A47	0.00
5	067203128142739c	abatement	greenhouse gases	A47	0.25
6	061d17f04be2d1cf	abatement	increased rate	A47	0.25
7	e1f44e48399a2027	abatement	measurement level	A47	0.25
8	0a425937a3e86d10	abatement	minimising sounds	A47	0.50
9	ef2d4c2e6bbb208d	abatement	mixing core materials	A47	0.25

Figure 5: *Training dataset*

code	title	section	class
A01	AGRICULTURE; FORESTRY; ANIMAL HUSBANDRY; HUNTI...	A	1.0
A21	BAKING; EDIBLE DOUGHS	A	21.0
A22	BUTCHERING; MEAT TREATMENT; PROCESSING POULTRY...	A	22.0
A23	FOODS OR FOODSTUFFS; TREATMENT THEREOF, NOT CO...	A	23.0
A24	TOBACCO; CIGARS; CIGARETTES; SIMULATED SMOKING...	A	24.0
...	...	...	...
H05	ELECTRIC TECHNIQUES NOT OTHERWISE PROVIDED FOR	H	5.0
H99	SUBJECT MATTER NOT OTHERWISE PROVIDED FOR IN T...	H	99.0
Y02	TECHNOLOGIES OR APPLICATIONS FOR MITIGATION OR...	Y	2.0
Y04	INFORMATION OR COMMUNICATION TECHNOLOGIES HAVI...	Y	4.0
Y10	TECHNICAL SUBJECTS COVERED BY FORMER USPC	Y	10.0

code	title	section
A	HUMAN NECESSITIES	A
B	PERFORMING OPERATIONS; TRANSPORTING	B
C	CHEMISTRY; METALLURGY	C
D	TEXTILES; PAPER	D
E	FIXED CONSTRUCTIONS	E
F	MECHANICAL ENGINEERING; LIGHTING; HEATING; WEA...	F
G	PHYSICS	G
H	ELECTRICITY	H
Y	GENERAL TAGGING OF NEW TECHNOLOGICAL DEVELOPME...	Y

Figure 6: *Context code titles provided by CPC. There are 9 sections and 136 section-class combinations.*

For this study, the input of the model will be anchor and target (words and word phrases) together with the context abbreviation that each represents a classification term categorized by patent experts manually. The score will be the output. In order to create this model, the pre-trained Decoding-enhanced BERT with disentangled attention (DeBERTa) natural language model was used. To have a human-like understanding of a language, the model should be

trained on large datasets with a very big model. This work requires a huge amount of time and computing resources. So using a pre-trained language model is critical: employing the trained weights and building on top of already trained weights reduces the overall compute cost and carbon footprint. They provide high accuracy with less training time compared to custom-build models and it is easier to implement.

The input data was prepared in 3 different formats; (1) only containing the anchor and target strings, (2) containing the context as an abbreviation together with the anchor and target, and (3) containing the context as a string description provided by CPC together with the anchor and target. So, the effect of having a topic on understanding the phrases can be observable. In order to avoid overfitting and unlucky data separations, K-fold cross-validation was used on the training dataset. The input dataset is tokenized so that it can be used for pre-trained models. The auto-tokenizer from the pre-trained model ensures tokenization is efficient and fast parallel with the different pre-trained models used. Also with padding and truncation, the same lengthed tokens are provided. The score has introduced the model as labels since it is not a continuous value. The created input was used to fine-tune to update the parameter weights of a pre-trained language model for the specific task.

(1) df['inputs'] = df.anchor + [sep] + df.target

(2) df['inputs'] = df.context + [sep] + df.anchor + [sep] + df.target

(3) df['inputs'] = df.title + [sep] + df.anchor + [sep] + df.target

The correlation between predicted and actual similarity scores is the main metric that we try to approach for measuring the model improvement. The competition's results were evaluated on the Pearson correlation coefficient between the predicted and actual similarity scores [16] when submitted. For this reason, the Pearson evaluation method was suggested to measure the degree of relationship between two variables. It is a linear correlation, varying from 1 to -1,

calculated with the ratio of correlation of two variables and the product of their standard deviations.

Results and Discussion

For this study, DeBERTaV3 pre-trained models [17] with different parameter sizes were used. The selected models are microsoft/deberta-v3-xsmall, microsoft/deberta-v3-small and microsoft/deberta-v3-base. The models already contain a lot of information about the language. They trained with 128k vocabulary and 22, 44, and 86 backbone parameters respectively.

Model	Vocabulary(K)	Backbone Parameters(M)	Hidden Size	Layers
DeBERTa-V3-Large[2]	128	304	1024	24
DeBERTa-V3-Base[2]	128	86	768	12
DeBERTa-V3-Small[2]	128	44	768	6
DeBERTa-V3-XSmall[2]	128	22	384	12

Figure 7: *Fine-tuning on natural language understanding tasks (SQuAD 2.0 and MNLI) (https://github.com/microsoft/DeBERTa)*

The models finetuned with the prepared dataset explained in the previous section and parameters were determined using 4-fold cross-validation with seed 42, group by the anchor. For the whole process, the epoch number is fixed to 4. The authors [17] recommend using only 10 epochs of training for fine-tuning DeBERTaV3 on the given NLP task in figure 7. However, there is a general sense to use 2-4 epochs and our input datasets do not contain long texts so they do not require many epochs. Weight decay for the regularisation term was determined as 0.01 with a learning rate of 8e-5. The training and evaluation batch size for each device is 64 for small

models and 32 for the base model to fit the free GPU provided by Kaggle. As explained in the previous section, compute metrics on the training data is Pearson correlation.

Epoch	Training Loss	Validation Loss	Pearson
1	No log	0.027643	0.780082
2	0.046200	0.024055	0.805587
3	0.021500	0.029485	0.805461
4	0.014000	0.024695	0.808442
5	0.010100	0.025080	0.809332
6	0.007800	0.025769	0.811355
7	0.007800	0.025145	0.808337
8	0.006300	0.025190	0.808423

Figure 8: *Example for 8 epochs of 'microsoft/deberta-v3-small' model with input-set(1)*

Using this project, it will be demonstrated that word or phrase meanings may be learned with pre-trained systems by having a general understanding of the issue, reducing the need for a long written explanation. Table 1 demonstrates the prediction results of different models with changing the input information. First, only the phrases to be associated (anchor and target) and their corresponding scores showing the degree of similarity are given as input. The second one includes the code of the universal category the phrases are included in (context). And the final one contains the scope of the codes (title) with the phrases. So by going right through the table, we are adding more information. The models are explained at the beginning of the chapter. As you move down the table, the number of backbone parameters increases thus more features are extracted from the source that the models trained.

The results in table 1 show that increasing the model size helped to improve results by around 3%. On the other hand, the main goal of this study is to understand the effect of a short context in which phrases' similarities are evaluated. As seen from the table, moving left to right, the results are growing steadily and supportively but the overall increase is around 1%.

Table 1: Pearson correlation results between predicted and true similarity scores.

	Only anchor and Target (1)	Anchor and target with context (2)	Anchor and target with the title (3)
microsoft/deberta-v3-xsmall	0.7974	0.7976	0.8043
microsoft/deberta-v3-small	0.8156	0.8179	0.8212
microsoft/deberta-v3-base	0.8189	0.8223	0.8301

As assumed at the beginning of the study, adding short texts to model improve the results for each model that I tried. In order to fully understand the significance of only adding a short text is possible with comparing it to the model provided a paragraph of description instead of the short text description. As intuition, providing more data will provide better results but it requires more computational power and resources. However, the source patent documents of the data do not available.

For future work, larger models may be tried with longer but handlable descriptions if the data is available. Also trying different tokenization strategies may help like grouping the same anchors with many targets and their scores to understand the relationship between the targets and create a weight distribution for scores accordingly. This strategy does not apply to the current situation since the real test dataset is closed to the public and it is not possible to change its given format.

Although the context descriptions added to the model other than phrases showed successful results, they could not show the improvement provided by the increase in model parameters. Another interesting research might be to explore the limits of success achieved by the increase in model parameters.

Conclusion

Due to the complex structure of languages, words and word structures can have very different meanings from the individual meanings of words when they are brought together in different contexts. In this study, it is investigated whether the semantic proximity of phrases is easier to predict when the contexts in which they are used are presented. For this purpose, pre-trained models of different sizes were used, and each model was given phrases in 3 different formats to determine their semantic similarities; (1) only containing the anchor and target strings, (2) containing the context as an abbreviation together with the anchor and target, and (3) containing the context as a string description provided by CPC together with the anchor and target. Results showed that similarity score estimation improved when phrases were presented with the context in which they were used. However, this increase is lower than the increase obtained by magnifying the model parameters.

References:

1. Trappey, A.J., Trappey, C.V. & Wu, CY. Automatic patent document summarization for collaborative knowledge systems and services. J. Syst. Sci. Syst. Eng. 18, 71–94 (2009).
2. Chen, L., Xu, S., Zhu, L. et al. A deep learning-based method for extracting semantic information from patent documents. Scientometrics 125, 289–312 (2020).
3. Amy J.C. Trappey, Charles V. Trappey, Jheng-Long Wu, Jack W.C. Wang, Intelligent compilation of patent summaries using machine learning and natural language processing techniques. Advanced Engineering Informatics, Volume 43 (2020).
4. D. W. Otter, J. R. Medina, and J. K. Kalita, "A Survey of the Usages of Deep Learning for Natural Language Processing," in IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 2, pp. 604-624, Feb. 2021
5. Qiu, X., Sun, T., Xu, Y. et al. Pre-trained models for natural language processing: A survey. Sci. China Technol. Sci. 63, 1872–1897 (2020).
6. Mikolov, Tomas, et al. "Distributed representations of words and phrases and their compositionality." Advances in neural information processing systems 26 (2013).
7. Mansoor, Muhammad & Rehman, Zahoor & Shaheen, Muhammad & Khan, Muhammad & Habib, Mohamed. (2020). Deep Learning based Semantic Similarity Detection using Text Data. Information Technology And Control. 49.
8. Tom Kenter, Alexey Borisov, Christophe Van Gysel, Mostafa Dehghani, Maarten de Rijke, and Bhaskar Mitra. 2018. Neural Networks for Information Retrieval. In WSDM 2018: WSDM 2018: The Eleventh ACM International Conference on Web Search and Data Mining , February 5–9, 2018, Marina Del Rey, CA, USA. ACM, New York, NY, USA, 2 pages
9. He, Hua and Jimmy J. Lin. “Pairwise Word Interaction Modeling with Deep Neural Networks for Semantic Similarity Measurement.” NAACL (2016).
10. S. Kale, E. Hazan, F. Cao, and J. P. Singh, "Analysis and algorithms for content-based event matching," 25th IEEE International Conference on Distributed Computing Systems Workshops, 2005, pp. 363-369, doi: 10.1109/ICDCSW.2005.40.
11. Guide to the International Patent Classification - WIPO 2022
12. Mikolov, Tomas, et al. "Efficient estimation of word representations in vector space." arXiv preprint arXiv:1301.3781 (2013).
13. 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.
14. Sarica, Serhad, Jianxi Luo, and Kristin L. Wood. "TechNet: Technology semantic network based on patent data." Expert Systems with Applications 142 (2020): 112995.
15. Schomacker, Thorben, and Marina Tropmann-Frick. "Language Representation Models: An Overview." Entropy 23.11 (2021): 1422.
16. https://www.kaggle.com/competitions/us-patent-phrase-to-phrase-matching/overview/evaluation

17. He, Gao, et al. “DeBERTaV3: Improving DeBERTa using ELECTRA-Style Pre-Training with Gradient-Disentangled Embedding Sharing” arXiv:2111.09543 (2021)