An Empirical Comparison of Semantic Similarity Methods for Analyzing down-streaming Automatic Minuting task

Aditya Upadhyay[&], **Aakash Bhatnagar**[1], **Nidhir Bhavsar**[$], **Muskaan Singh**[#] and **Petr Motlicek**[#]
[&] CSED, Thapar Institute of Engineering and Technology, India
[1] Boston University, Boston, Massachusetts
[$]University of Potsdam, Potsdam, Germany
[#] Speech and Audio Processing Group, IDIAP Research Institute, Martigny, Switzerland
`adityaupadhyay1912@gmail.com`, `aakash07@bu.edu`,
`bhavsar@uni-potsdam.de`, `(msingh,petr.motlicek)@idiap.ch`

Abstract

Automatic Minuting consists of automatically creating minutes from multiparty meeting transcripts. In this paper, we solve two relevant problems of this domain (1) given a pair of meeting transcript and minute, the task is to identify whether the minutes belongs to the transcript. (2) given a pair of minutes, the task is to identify whether the two minutes belong to the same or different meetings. These challenging problems are important as we want to uncover how minutes created by two different persons for the same meeting may differ in content and coverage. The proposed system leverage off-the-shelf semantic similarity techniques which provides with a score of similarity indicating a measure of how close the two text are to each other in meaning. In performance analysis, we broadly formulate three categories with the best performers in each (1) in lexical summarization DOC2VEC (2) in machine learning (3) in deep transformer architectures. We evaluate each of our proposed approaches on the basis of Accuracy. For lexical summarization, Doc2Vec achieves 90% and 51% accuracy, in machine learning, random forest achieves 91% and 85% accuracy and in deep learning, STSB-BERT-Large achieves 94% and 81% accuracy in transcript-minutes and minutes-minutes respectively.

1 Introduction

Due to differences in the style of minuting there are two important challenges,(i) identify if the minutes and the transcript are from the same meeting. (ii) identify if two minutes are from the same meeting (which are taken by different note takers). In this paper, we focus to solve this novel problem and see to what extent this decision can be carried out automatically. The novelty of our research is to *examine the subjectivity associated with the minuting exercise*. Minuting is a challenging task [1], and even more difficult is identify meetings similarity on similar topics with (1) similarity of discussed content and anchor points like named entities e.g. in recurring meetings of the same project on the one hand, and the differences in the style of minuting on the other hand. (2) some minutes do not capture the central points in the meeting because the external scribes did not understand the context correctly and created minutes which miss significant issues discussed in the meeting or are simply too short.

2 Background

Semantic Similarity is a Natural Language Processing (NLP) task that consists of measuring the similarity between two texts in a quantitative manner. Measuring semantic similarity between two texts serves an important role in various NLP tasks such as information retrieval, text summarization, text classification, essay evaluation, machine translation, and question answering [2]. One of the major challenges in the semantic similarity problem is the complex nature of semantics, for instance the words *car* and *gasoline* are definitely more closely related than the words *car* and *bike*, but the latter pair is more similar than the former. Some of the earliest methods used to solve the semantic similarity problem involved embedding words using systems of taxonomy that paired similar words together in trees, such as the WordNet taxonomy. The major issue in this approach was that it relied on the assumption that links in the taxonomy represented similar distances which was not always the case.

3 Related Work

There are various different methods to measure semantic similarity. These include deep learning approaches, machine learning approaches and standard algorithmic approaches. [2] provides a survey of various different methods to semantic similarity along with datasets and semantic distance measures. Within machine learning approaches regression is a popular technique to measure semantic similarity regression is a predictive modelling technique that is used to obtain the relation between the target and the input features. [3] compares various different regression models on the SemEval dataset. They found Boosting to have the best performance when compared to methods such as Bagging, Multi Linear, RPart, Random Forest, and SVM algorithms. [4] proposed using pairwise word interactions in order to find the context based correlation as neural based approaches seem to have difficulty in finding word level similarity. The model performance was texted using news articles, headlines and other such datasets and the model was found to perform very well compared to standard neural networks.

[5] proposed an improved version of the Bi-LSTM model based on the Siamese network architecture. The model was trained on a QA dataset consisting of 100,000 question pairs, 10 fold cross validation was used as a loss metric and the model was found to perform significantly better than other models achieving an accuracy of 84.87%. [6] implemented SVM with CNN and Siamese Recurrent architecture for RNN on a QA dataset. They found the CNN with SVM doesn't correctly assess if the statement has some image and RNN has a Vanishing Gradient problem. [7] also proposed 7 different variants of the RNN architecture using the SICK dataset and the STS2017 dataset. The best performance was achieved by a model that contains a single GRU cell.

Embeddings based neural networks are also widely used to the task of semantic similarity.[8] compares various embeddings based models trained on 1.7 million articles from the PubMed Open Access dataset. These models were tested on a biomedical benchmark (BIOSSES) set that contains 100-sentence pairs. They found Paragraph Vector Distributed Memory algorithm to outperform all other models achieving a correlation of 0.819. [9] compared a CNN model with six other models and used the LIME algorithm to identify the keywords and improve model performance. The other approaches based on deep learning methods [10, 11, 12, 13, 14, 15, 16, 17, 18, 18, 19, 20, 21, 22, 23, 24, 25, 26] proposed by various researchers improve semantic similarity for different applications.

4 Methods

We perform an empirical comparison using off-the-shelf methods of semantic similarity to downstream novel task of determining whether minutes belong to a meeting and whether two sets of minutes belong to a meeting. We formulate two broad categories, namely (1) lexical similarity techniques (2) machine learning based similarity techniques (3) deep transformer architectures.

The lexical similarity methods compares word lengths and character-wise similarity by embedding the contents of the input texts into vectors and then determining the semantic distance between those vectors. In our work we use,

- Bag-Of-Words (BoW) [27] was one of the first methods to embed data for text classification ever developed. It converts a document into a set of words keeping the frequency of each word as a feature in the set. This frequency is used as the embedding for the term.

- Doc2Vec is an implementation of paragraph embedding that was initially proposed in [28]. paragraph embedding uses a log-probability function to obtain the probability of each word in the input text and then uses a function such as softmax to classify the word into a vector. Doc2Vec uses hierarchial softmax to embed the input text.

- Named Entity Similarity [29] is a method that extracts the named entities in a document (real world objects) and embeds them based on the type of entity they correspond to, for instances Apple is an organization while U.K. is a Geopolitical Entity.

- Keyword Similarity Words that seem important or representative of the text are extracted as keywords using the BM25 Ranking Function proposed in [30] to extract the importance of a word. The sets of keywords extracted are then compared to determine similarity.

- Cosine Similarity measures the similarity between two vectors via inner product. It is measured by the cosine of the angle between two vectors and determines whether two text article/documents are pointing in roughly the same direction. For computing the similarity between the text documents we considered using the cosine similarity pairwise metric by sklearn.

$$\text{similarity } = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} \tag{1}$$

- Summarized Keyword Similarity, entire texts are summarized using the BM25 Ranking Function to extract important sentences. Keywords are then extracted using the same method to get summarized keywords for each. These summarized keywords are compared to determine similarity.

- Summarization based Named Entity Similarity, entire texts are summarized using the BM25 Ranking Function to extract important sentences. Named entities are then extracted and embeded based on the entity they correspond to. These embeddings are compared to determine similarity.

- SequenceMatcher is a class that is available under the difflib Python package[1]. Main objective behind *SequenceMatcher* is to find the longest contiguous matching sub-sequence LCS with no "irrelevant" elements. Irrelevant are the characters that we don't want the algorithm to match, like blank lines in ordinary text files, etc. This metric does not yield minimal edit sequences, but does tend to yield matches that logically seems appropriate.

- Jaccard Similarity, measure the similarity of two meeting minutes in terms of their context, i.e. how many common words there are compared to the total number of words. Here J is the Jaccard distance calculated via the distinct word present in set A and B.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{2}$$

These methods pose challenge, to capture the syntactic and semantic meaning of the words in a text and therefore cannot effectively measure semantic similarity. For instance, "I have a weak disposition" and "I often get sick" have the same meaning but due to a lack of similar words would not measure a high semantic similarity using these methods. Additionally, synonyms and ambiguous words (have multiple meanings in different contexts).

In machine learning, we implement Support Vector Machine (SVM) and the Random Forest Classifier.

please add about SVM and random forest

Deep transformer architectures, embed text into vectors to measure semantic similarity. These models are trained on massive corpora of words and take syntactic meaning as well as context into account when embedding sentences. This allows them to make relatively accurate measures of semantic similarity when compared to traditional approaches.

In our work, we use,

- Universal Sentence Encoder is a sentence encoder that was developed by researchers at google in [31]. The encoder was trained on unsupervised data collected from various sources including Wikipedia, various web news and discussion forums. The unsupervised data was augmented with training on supervised data from the Stanford Natural Language Inference (SNLI) corpus [32].

- BERT (Bidirectional Encoder Representations from Transformers) is a bidirectional transformer model proposed in [33]. BERT was trained using a combination of masked language modeling objective and next sentence prediction (NSP) on a large corpus comprising the Toronto Book Corpus and Wikipedia. In our work, we experiment with BERT namely stsb-bert-large [2], stsb-bert-base [3], nli-bert-large [4], nli-bert-large-max-pooling [5], nli-bert-large-cls-pooling [6], nli-

[1] https://docs.python.org/3/library/difflib.html

[2] https://huggingface.co/sentence-transformers/stsb-bert-large

[3] https://huggingface.co/sentence-transformers/stsb-bert-base

[4] https://huggingface.co/sentence-transformers/nli-bert-large

[5] https://huggingface.co/sentence-transformers/nli-bert-large-max-pooling

[6] https://huggingface.co/sentence-transformers/nli-bert-large-cls-pooling

bert-base-max-pooling [7],nli-bert-base [8], nli-bert-base-cls-pooling [9]

- RoBERTa (Robustly optimized BERT approach) [34] uses the same architecture as BERT but modifies the pre-training step. Specifically, RoBERTa is trained with dynamic masking, FULL-SENTENCES without NSP loss, large mini-batches and a larger byte-level BPE. We experiment with stsb-roberta-large[10], stsb-roberta-base[11], nli-roberta-large[12], nli-roberta-base[13].

- DistilBERT (Distilled BERT) [35] is a fast and light variant of BERT. It is trained 40% less parameters than BERT, runs 60% faster while preserving over 95% of BERT's performances as measured on the GLUE language understanding benchmark. We use stsb-distilbert-base[14], paraphrase-distilroberta-base-v1[15], nli-distilbert-base[16], nli-distilbert-base-max-pooling[17].

- XLM (cross-lingual Language Model) [36] is a transformer based model that is trained on Next Token Prediction (causal language modeling (CLM) objective), masked language modeling (MLM) objective and a Translation Language Modeling (TLM) object.In our work, we use paraphrase-xlm-r-multilingual-v1 model[18].

5 Experimental Setup

In this section, we describe our experimental setup for the empirical comparison of off-the-shelf methods for our novel down-streaming task. We describe dataset in Section 5.1 and hyperparameter in Section 5.2.

5.1 Dataset Details

The dataset for determining whether minutes belong to a meeting, consists of pairs of transcripts and minutes that are labelled either True or False depending on whether they were derived from the same conversation or not, i.e. True implies the minutes match the transcript and vice versa. The other problem, consists of pairs of minutes that are labelled True of False depending on whether they belong to the same meeting or not. We have used the English and Czech datasets for both tasks. Both these datasets are strongly imbalanced with only around 15% of the pairs belonging to the True class in each case.

5.2 Hyperparameters

We use similar hyperparameters of all the transformer models, with sequence length of 128, word embedding dimension as 1024, drop_out rate of 0.1, hidden_size of 1024, initializer_range as 0.02, intermediate_size of 4096, layer_norm 1e-05 epissilon value, and max_position_embeddings of 514. There are a few more parameters such as pooling type that are different for different models. Some use max pooling while others use mean or CLS pooling. These hyperparameters were picked to tune the models to improve performance. The models were trained on their respective datasets using these hyperparameters

To perform this classification, the similarity values are produced on the embedding produced by a pre-trained model, and then a threshold is used to achieve the binary classification. The pretrained model used is ”bert-base-nli-mean-tokens” provided by hugging face. In this model, BERT-base has been used, which creates the dense vectors containing 768 values. These 768 values contain our numerical representation of a single token — which we can use as contextual word embedding. Some other hyperparameters for this model

[7] https://huggingface.co/sentence-transformers/nli-bert-base-max-pooling

[8] https://huggingface.co/sentence-transformers/nli-bert-base

[9] https://huggingface.co/sentence-transformers/nli-bert-base-cls-pooling

[10] https://huggingface.co/sentence-transformers/stsb-roberta-large

[11] https://huggingface.co/sentence-transformers/stsb-roberta-base

[12] https://huggingface.co/sentence-transformers/nli-roberta-large

[13] https://huggingface.co/sentence-transformers/nli-roberta-base

[14] https://huggingface.co/sentence-transformers/stsb-distilbert-base

[15] https://huggingface.co/sentence-transformers/paraphrase-distilroberta-base-v1

[16] https://huggingface.co/sentence-transformers/nli-distilbert-base

[17] https://huggingface.co/sentence-transformers/nli-distilbert-base-max-pooling

[18] https://huggingface.co/sentence-transformers/paraphrase-xlm-r-multilingual-v1

include the non-linear activation function (function or string) in the encoder and pooler as ’gelu’, the number of attention heads for each attention layer in the transformer encoder as 12 and the standard deviation of the truncated-normal-initializer for initializing all weight matrices as 0.02. After this model produces the embeddings, cosine similarity (measured similarity between two feature vectors by capturing the document’s orientation and not the magnitude, unlike the Euclidean distance) is measured, and the final similarity values are produced. Many threshold values are checked to minimize the mismatching of actual binary classifications and the generated binary classifications. The final threshold value is chosen by 0.65.

A pretrained model was used to obtain the sentence embedding. Then a similarity metric was used to get the similarity values, and finally, a threshold value was obtained to get the final binary classification. The pretrained model used is ”paraphrase-distilroberta-base-v1’, which is a ’DistilBERT-base-uncased’model fine-tuned on a large dataset of paraphrase sentences. This RoBERTa-based sentence representation model has been trained to produce meaningful sentence embedding for similarity assessment and retrieval tasks. It uses a vector length of 768 for the sentence embeddings. Some other hyperparameters for this model include the non-linear activation function (function or string) in the encoder and pooler as ’gelu’, the number of attention heads for each attention layer in the transformer encoder as 12 and the standard deviation of the truncated-normal-initializer for initializing all weight matrices as 0.02. After this model produces the embeddings, cosine similarity is measured, and the final similarity values are produced. Many threshold values are checked to minimize the mismatching of actual binary classifications and the generated binary classifications. The final threshold value is chosen by 0.65. The final scores yield an accuracy of 79.8%.

6 Results and Analysis

All the above listed models were tested on the Automin Dataset Minutes-Transcript and Minutes-Minutes using cosine distance as a measure of semantic distance. The testing was carried out on an Nvidia K80 with 2496 CUDA cores operating at 4.1 TFLOPS with 12 GB of primary memory and a hyper-threaded Intel Xeon processor with 2 cores operating at 2.3 GHz. The compute times of each approach can be found in Figure 1.

In our results, we perform quantative evaluation using accuracy and we also vouch for qualitative analysis. The results of the tests on lexical analysis methods can be found in Table 1. It can be observed that Keyword similarity had by far the best performance with Summarization Keyword Similarity being a close second. Summarization based Named Entity Similarity had the best performance on minutes-transcript but performed poorly on minutes-minutes, this disparity can be attributed to the nature of the datasets. The results of the tests performed on the machine learning algorithms can be found in Table 3. The results of the tests on Transformer Based Deep Learning models can be found in Figure 2. In Figure 1 we can observe the computational times for the different deep learning models. Snippets of the datasets for true positive, true negative, false negative and false positive results from the stsb-bert-base model can be found in Figure 3, Figure 4, Figure 5 and Figure 6 respectively.

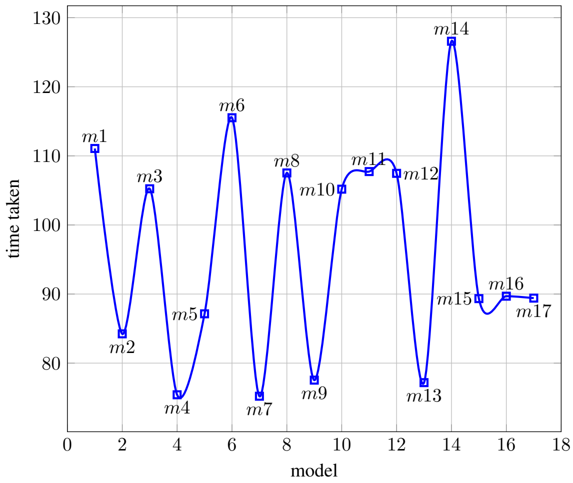

Figure 1: m1: stsb-roberta-large, m2: stsb-roberta-base, m3: stsb-bert-large, m4: stsb-distilbert-base, m5: stsb-bert-base, m6: paraphrase-xlm-rmultilingual-v1, m7: paraphrasedistilrobertabase-v1, m8: nli-bert-large, m9: nli-distilbert-base, m10: nli-roberta-base, m11: nli-bert-large-maxpooling, m12: nli-bert-large-clspooling, m13: nli-distilbert-basemax-pooling, m14: nli-roberta-base, m15: nli-bert-base, m16: nli-bert-base-cls-pooling.

7 Conclusion

Based on the observations drawn from the tests performed, we can conclude that Transformer

Table 1: This table shows the results of all the lexical similarity methods for semantic similarity

Approach	Accuracy (Task B)	Accuracy (Task C)
DOC2VEC	0.9089577851	0.51258137
Named Entity Similarity	0.04519868388	0.05636713945
Keyword Similarity	0.7957055804	0.68533914
Summarization Keyword Similarity	0.523513454	0.4626331746
Summarization based Named Entity Similarity	0.02417613166	0.7510776139
Feature Engineering (Bag of Words)	0.21272047	0.01208951125
Universal Sentence Encoder	0.2644637739	0.4936408219

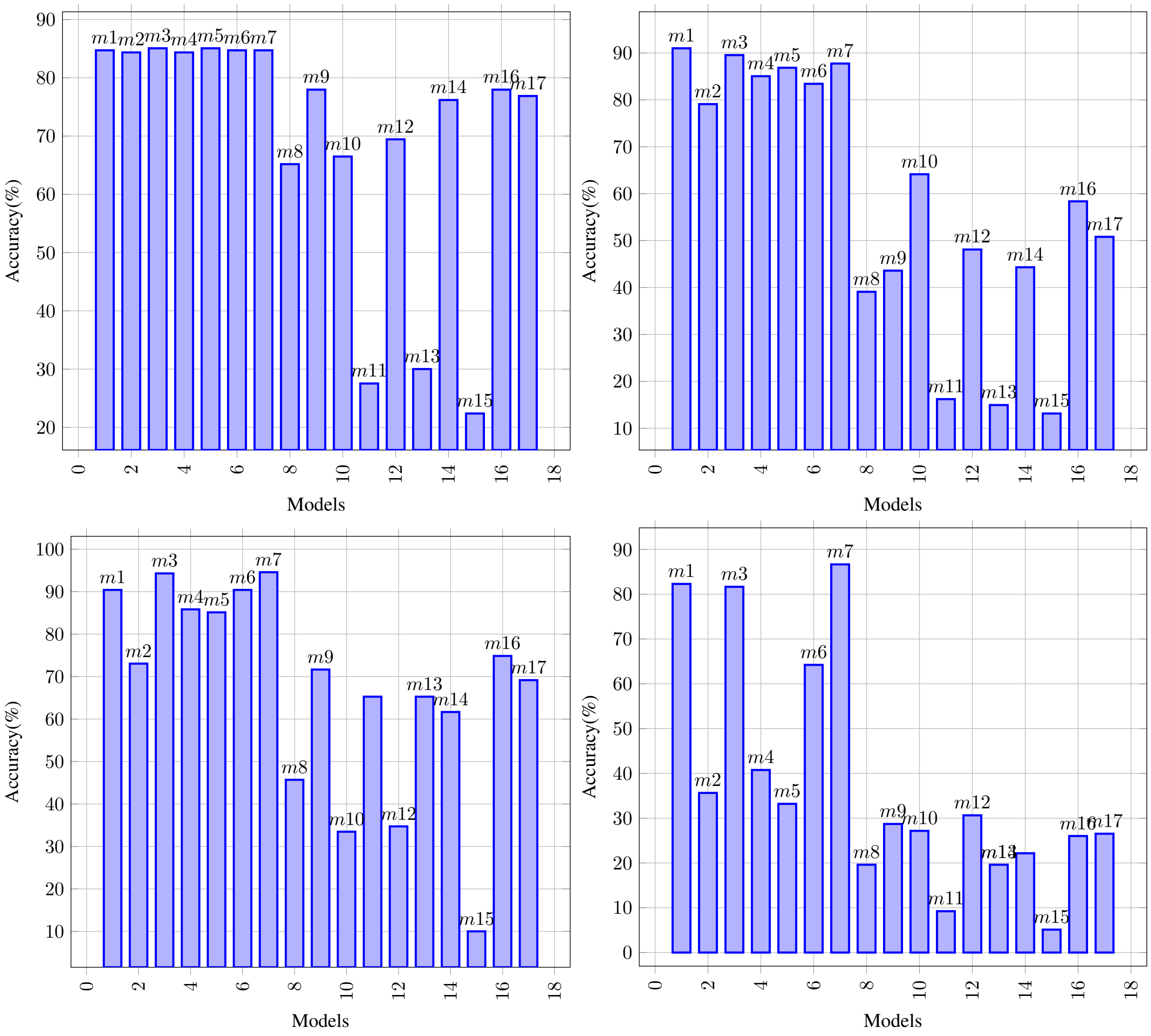

Figure 2: The first row contains the evaluation score of english language, while the bottom two graphs contain evaluation scores of czech m1: stsb-roberta-large, m2: stsb-roberta-base, m3: stsb-bert-large, m4: stsb-distilbert-base, m5: stsb-bert-base, m6: paraphrase-xlm-rmultilingual-v1, m7: paraphrasedistilrobertabase-v1, m8: nli-bert-large, m9: nli-distilbert-base, m10: nli-roberta-base, m11: nli-bert-large-maxpooling, m12: nli-bert-large-clspooling, m13: nli-distilbert-basemax-pooling, m14: nli-roberta-base, m15: nli-bert-base, m16: nli-bert-base-cls-pooling.

based models perform far better than lexical analysis methods. It can be observed that models based on RoBERTa, particularly roberta-large have the best performance on both minutes-transcript and minutes-minutes on the English datasets while the distilroberta-base model trained on the paraphrase dataset had the best performance on transcript-minutes and minutes-minutes on the

	Classifier	Acc	Pre	Rec	F1
Task B	Random-Forest	**0.91**	**0.71**	**0.62**	**0.66**
	SVM	0.88	0.65	0.40	0.49
Task C	Random-Forest	**0.85**	**0.42**	**0.61**	**0.5**
	SVM	0.77	0.26	0.53	0.35

Table 2: Describes the various machine learning classification approaches

Dataset	True tag	False tag	**Total**
Task B	115	731	**846**
Task C	74	660	**734**

Table 3: Class-wise distribution of Data.

Czech dataset. Models based on BERT and models trained on NLI in general performed poorly on both tasks in both languages. For lexical summarization, Doc2Vec achieves 90% and 51% for respectively. In machine learning, random forest achieves 91% and 85% and in deep learning STSB-BERT-Large outperforms all other with 94% and 81%

8 Acknowledgements

This work was supported by the European Union's Horizon 2020 research and innovation program under grant agreement No. 833635 (project ROXANNE: Real-time network, text, and speaker analytics for combating organized crime, 2019-2022).

(A) Meeting transcript segment:
(PERSON1) Yeah, that's, exactly. Ok. So Monday seminar setting. Screensharing. Yeah, yeah, so exactly. So the demo input, I think everybody I'm going through to do list. And I think that the set of the languages it should be all we have at the moment. But the main question is which should be the input language and in the call they [ORGANIZATION2] said the German would be ok for them. So I think we should be ready for for German source.
(PERSON9) Ok, nice.
(PERSON1) And like double check with them but ¡unintelligible¿ if they if they now realize that German is a bad idea then that's now a problem and we switch to English as the source.
(PERSON9) Ok. I'm going to adds this one. German as input and English as back up. Ok. Well, and translate it in all the available languages are we sure?
(PERSON1) Yeah, I would say so. Why not?It's, what would, what could be the problem?
(PERSON9) I don't know. Do we have a tested the machine translation, well the speech language translation starting from German to so all the other possible languages?
(PERSON1) So it goes via English of course. And what we have tested several times is from Czech into English and then from English into the other languages. So we have not tested-
(PERSON9) Ok.
(PERSON1) With German source, I agree. But I kind of trust the German to English [ORGANIZATION2] model. And I the the English to everything is the best that we have all the time. Like it's-
(PERSON9) Ok, ok.

(B) Meeting minutes Date: 2020/05/04
Attendees: [PERSON1], [PERSON9], [PERSON2]
Purpose of meeting: Demo preparations.
Summary of meeting:
[PERSON9], [PERSON1]
- choose [ORGANIZATION2] and [ORGANIZATION5] persons.
- From [ORGANIZATION2] is chosen [PERSON8], from [ORGANIZATION5] [PERSON8].
[PERSON9], [PERSON1]
- discuss demo input.
- German as input and English as back up.
- There are prepared some Youtube videos that are already consecutevely translated into Czech.

Cosine Distance 0.83442569631

Figure 3: An example of one of the sets in Task B where the minutes and transcript belong to the same meeting.

References

[1] Tirthankar Ghosal, Muskaan Singh, Anna Nedoluzhko, and Ondřej Bojar. Report on the SIGDial 2021 special session on summarization of dialogues and multi-party meetings (summdial). *ACM SIGIR Forum*, December 2021:1–17, 2021.

[2] Dhivya Chandrasekaran and Vijay Mago. Evolution of semantic similarity—a survey. *ACM Comput. Surv.*, 54(2), feb 2021.

[3] V Sowmya, K Kranthi Kiran, and Tilak Putta. Semantic textual similarity using machine learning algorithms. *International journal of current engineering and scientific research (IJCESR)*, pages 2393–8374, 2017.

[4] Hua He and Jimmy Lin. Pairwise word interaction modeling with deep neural networks for semantic similarity measurement. In *Proceedings of the 2016 conference of the north American chapter of the Association for Computational Linguistics: human language technologies*, pages 937–948, 2016.

[5] Zongkui Zhu, Zhengqiu He, Ziyi Tang, Baohui Wang, and Wenliang Chen. A semantic similarity computing model based on siamese network for duplicate questions identification. In *CCKS Tasks*, pages 44–51, 2018.

[6] J Ramaprabha, Sayan Das, and Pronay Mukerjee. Survey on sentence similarity evaluation using deep learning. In *Journal of Physics: Conference Series*, volume 1000, page 012070. IOP Publishing, 2018.

[7] Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. Semantic textual similarity with siamese neural networks. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1004–1011, 2019.

[8] Kathrin Blagec, Hong Xu, Asan Agibetov, and Matthias Samwald. Neural sentence embedding models for semantic similarity estimation in the biomedical domain. *BMC bioinformatics*, 20(1):1–10, 2019.

[9] Tao Zheng, Yimei Gao, Fei Wang, Chenhao Fan, Xingzhi Fu, Mei Li, Ya Zhang, Shaodian Zhang, and Handong Ma. Detection of medical text semantic similarity based on convolutional neural net-

(A) Meeting transcript segment:
(PERSON31) Mmm hm.
H- How bad eh, mistake was that with the deadline of ¡laugh¿ manual annotation for [PROJECT1]?
I think -
(PERSON31) Ehm -
Yeh, you aren´t really asking the right person here.I don´t know in in a sense that - After the annotations done, somebody got to analyze them.
And that person is hopefully (producer??) not me, so -
(PERSON9) Yeah. ¡laugh¿
(PERSON31) ¡laugh¿
Ehm -
And then somebody´s gonna try them up again.
That´s hopefully not me.
And that really needs to be done on time for the conference.
I think last time we had like the day before the conference.
(PERSON9) Yes, yes -
(PERSON31) So we don´t want to be any later than that.
(PERSON28) But the term - ¡parallel$_{s}peech >$
(PERSON9) But the conference is - ¡parallel$_{s}pech >$
(PERSON28) is on Saturday this Saturday.
(PERSON9) Yeah eh, and the conference is in November right?
(PERSON31) Yeah, eh in- eighteenth or so -
(PERSON9) Eighteenth, yes .
(PERSON31) So it´s is probably still time that we get the annotations done for the twentieth.
I mean we´re just just so late this year for -
(PERSON9) For everything.
So, so well, actually -
(PERSON31) ¡parallel$_{s}peech >$

(B) Meeting minutes [PROJECT1] internal Meeting
Date: 7. 9. 2020:
Attendance: [PERSON4], [PERSON13], [PERSON2],[PERSON11], [PERSON6]- Paraphrasing on
Quest:
– We should move it to "weaker" GPU (requested by PROJECT4)
– Can we do it before the end of the [PROJECT4] experiment? ("on the fly")
- [PROJECT3]:
– in contact with [ORGANIZATION2], they provide us with their multilingual data
- [PROJECT4]:
– [PERSON12] is leaving the project!
– [ORGANIZATION4]has System Demonstration track - do we want to participate?
- [PROJECT2]:
– people from [ORGANIZATION6] (name, contact???) are also working on the decoding
constraints (or factored translation?)
- [PERSON4] getting details from [PERSON7]

Cosine Distance 0.247577452

Figure 4: An example of one of the sets in Task B where the minutes and transcript do not belong to the same task.

(A) Meeting transcript segment:
(PERSON2) So [PERSON5] is probably already there right?
That was [PERSON7], right.
So [PERSON17] would [PERSON5] join would you know?
(PERSON15) Yes yes, he he ¡unintelligible¿ minutes.
(PERSON2) Okay.
In the minutes okay.
And you are listening, you can hear us right?
Okay, so [PERSON12] can h- is listening to to the Zoom call.
But he doesnt have the microphone
Because that was causing the loop yesterday.
So its only me who has a microphone.
So [PERSON7], v-.
How much time do you need? [PERSON7] is not here.
So [PERSON7] is also trying to set up what what he posted yesterday.
So that while watching the videos, a participants who do not speak Czech should be clicking to buttons, like how well they like the current subtitles, and it would be timestamped.
And we can then align it and see like where the problems it is going to be approximate because the sync of the video.
Is not perfect, as you know, but it still, will probably be useful to to identify the the to give us some measure of the overall usability of of that.
So a please look up in your emails, because [PERSON7] sent it this morning.
The [ORGANIZATION2] document and please sign up yourself if you can to the subtitle rating uh, documents, so.
Whoever is available.
Please write your name here that is what I 'm going to highlight.

(B) Meeting minutes Test session 20200515-1000 – instead of the real demo
• Credentials:
o [URL]
o Meeting ID: [NUMBER], Password: [PASSWORD]
• Meeting is already started, the room is available until 13.00.
• ([PERSON2] will need to leave at 12.00 at the latest)
• ([PERSON12] is available only on [ORGANIZATION3], his zoom is meant only for subtitling the zoom discussion)
• Agenda:
o Summary of worker instances involved ([PERSON12], just a recap, pasting aggtable here, highlighting it)
•
• Computer names need to be shown, too, so that we can check them for load.
o [PERSON2] giving dry run of the slides.
o Czech subtitling of both Czech sample videos.
• 3min [URL]
• 15 min from this: [URL]
o Possibly English subtitling with our segmenter for English videos and our zoom discussion.
o Czech subtitling of zoom discussion ([PERSON2] and any other Czech colleague present)
o [PERSON2] giving dry run of the closing slides.

Cosine Distance 0.3254856236

Figure 5: An example of one of the sets in Task B where the minutes and transcript do belong to the same meeting but the model was not able to label it correctly.

work. *BMC medical informatics and decision making*, 19(1):1–11, 2019.

[10] Richard Socher, Andrej Karpathy, Quoc V Le, Christopher D Manning, and Andrew Y Ng. Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, 2:207–218, 2014.

[11] Sergio Jimenez, Julia Baquero, and Alexander Gelbukh. Unal-nlp: Combining soft cardinality features for semantic textual similarity, relatedness and entailment. In *In Proceedings of SemEval 2014: International Workshop on Semantic Evaluation*. Citeseer, 2014.

[12] Alice Lai and Julia Hockenmaier. Illinois-lh: A denotational and distributional approach to semantics. In *SemEval@ COLING*, pages 329–334, 2014.

[13] Johannes Bjerva, Johan Bos, Rob Van der Goot, and Malvina Nissim. The meaning factory: Formal semantics for recognizing textual entailment and determining semantic similarity. 2014.

[14] Jiang Zhao, Tian Tian Zhu, and Man Lan. Ecnu: One stone two birds: Ensemble of heterogenous measures for semantic relatedness and textual entailment. semeval. 2014.

[15] Kai Sheng Tai, Richard Socher, and Christopher D Manning. Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075*, 2015.

[16] Yi Yang, Wen-tau Yih, and Christopher Meek. Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 2013–2018, 2015.

[17] Aliaksei Severyn and Alessandro Moschitti. Learning to rank short text pairs with convolutional

(A) Meeting transcript segment:
[ORGANIZATION2].
And that’s the leadership of work packages.
Because as [ORGANIZATION7] are like not- not a standalone, not- not a full partner here, uh, they cannot lead the work package, uh, on data.
Uh, so, uh, I thought that, uh, uh, it- it would be the- good that one of the research partners, uh, does this.
And, uh, we are responsible for the multilingual MT and management and for- uh, now for minuting as well.
Uh, uh, so, uh, like we have data and minuting kind of available, uh, for [ORGANIZATION2], uh, or [ORGANIZATION11], or we can do it, uh, uh, s- as well, but, uh-
yeah, so, uh, yeah, this is-
I don’t know if [ORGANIZATION11] has any opinion on this?
(PERSON14) So, uh, I think it would make more sense if, uh, [ORGANIZATION9] keeps the minuting part because you also, uh
(PERSON19) Okay.
(PERSON14) you- you’ve- meaning the, um, getting at the concept- you have the most involvement there.
(PERSON19) Yeah ¡unintelligible¿
(PERSON14) And I think, uh, it’s okay, to have the data work package. Either us or [ORGANIZATION2] ¡unintelligible¿ [ORGANIZATION2] what they- what their position is.
(PERSON11) Yeah
(PERSON19) Yeah, so I’ve talked to-
yeah, [PERSON3], [PERSON3], s- please, say this- say that.
(PERSON11) So yeah, I mean, uh, ¡unintelligible¿ of course you wrote most of it, so we can also keep the minuting, otherwise since K- uh, [ORGANIZATION11] has more on- on data, I would have suggested that [ORGANIZATION11] does data and we do minuting, but I’m fine with anything else too.

(B) Meeting minutes Meeting 16. 11. 2020:
• Attendance:
– [PERSON21], [PERSON3], [PERSON22], [PERSON23], [PERSON26]
- Paraphrasing
– Deadline for student - February
– Prepare the data (provided by [PERSON16]) by the end of the month
– [PERSON23] will try translating them by the TF model
– [PERSON26] will train parent in [PROJECT1]NMT
– Questions
i. ([PERSON26]) I got model description from [PERSON16]. Still waiting for training reproduction scripts for easy training setup.
ii. We also need to evaluate the performance loss cause by compression
iii. Can we distill paraphrasing directly? By training direct paraphrasing (student) model without pivoting?
- [PROJECT5]
– [PERSON23] is processing the segments provided by [ORGANIZATION7]
– waiting for [ORGANIZATION7] ([PERSON18])
– trying to submit paper to [ORGANIZATION1]
- [PROJECT2] (data augmentation)
– aiming for a workshop rather than [ORGANIZATION1]
- [PROJECT6]
– [PERSON22] is training an already constrained model
— comparing it with constrained decoding on inflected languages
– Next step is focus on constraining in inflected languages
- Implementation

Cosine Distance 0.77585155236

Figure 6: An example of one of the sets in Task B where the minutes and transcript do not belong to the same meeting but the model was not able to label it correctly.

deep neural networks. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 373–382, 2015.

[18] Basant Agarwal, Heri Ramampiaro, Helge Langseth, and Massimiliano Ruocco. A deep network model for paraphrase detection in short text messages. *Information Processing & Management*, 54(6):922–937, 2018.

[19] Rafael Ferreira, George DC Cavalcanti, Fred Freitas, Rafael Dueire Lins, Steven J Simske, and Marcelo Riss. Combining sentence similarities measures to identify paraphrases. *Computer Speech & Language*, 47:59–73, 2018.

[20] Yushi Homma, Stuart Sy, and Christopher Yeh. Detecting duplicate questions with deep learning. In *Proceedings of the International Conference on Neural Information Processing Systems (NIPS*, 2016.

[21] Jiangping Huang, Shuxin Yao, Chen Lyu, and Donghong Ji. Multi-granularity neural sentence model for measuring short text similarity. In *International Conference on Database Systems for Advanced Applications*, pages 439–455. Springer, 2017.

[22] Jyun-Yu Jiang, Mingyang Zhang, Cheng Li, Michael Bendersky, Nadav Golbandi, and Marc Najork. Semantic text matching for long-form documents. In *The World Wide Web Conference*, pages 795–806, 2019.

[23] Ankur P Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*, 2016.

[24] Zhiguo Wang, Haitao Mi, and Abraham Ittycheriah. Sentence similarity learning by lexical decomposition and composition. *arXiv preprint arXiv:1602.07019*, 2016.

[25] Wenpeng Yin and Hinrich Schütze. Convolutional neural network for paraphrase identification. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 901–911, 2015.

[26] Wenpeng Yin, Hinrich Schütze, Bing Xiang, and Bowen Zhou. Abcnn: Attention-based convolutional neural network for modeling sentence pairs. *Transactions of the Association for Computational Linguistics*, 4:259–272, 2016.

[27] Yin Zhang, Rong Jin, and Zhi-Hua Zhou. Understanding bag-of-words model: a statistical framework. *International Journal of Machine Learning and Cybernetics*, 1(1):43–52, 2010.

[28] Quoc V. Le and Tomás Mikolov. Distributed representations of sentences and documents. *CoRR*, abs/1405.4053, 2014.

[29] Jiahui Liu and Larry Birnbaum. Measuring semantic similarity between named entities by searching the web directory. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI’07)*, pages 461–465, 2007.

[30] Federico Barrios, Federico López, Luis Argerich, and Rosa Wachenchauzer. Variations of the similarity function of textrank for automated summarization. *CoRR*, abs/1602.03606, 2016.

[31] 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. Universal sentence encoder. *CoRR*, abs/1803.11175, 2018.

[32] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. *CoRR*, abs/1508.05326, 2015.

[33] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[34] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.

[35] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. *CoRR*, abs/1910.01108, 2019.

[36] Guillaume Lample and Alexis Conneau. Cross-lingual language model pretraining. *CoRR*, abs/1901.07291, 2019.