ΟΙΚΟΝΟΜΙΚΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΑΘΗΝΩΝ

ATHENS UNIVERSITY
OF ECONOMICS
AND BUSINESS

School of Information Sciences and Technology
Department of Informatics
Athens, Greece

Master Thesis
in
Computer Science

Exploring evaluation methods for automatically generated summaries

Stratos Xenouleas

Supervisors: Prodromos Malakasiotis

Marianna Apidianaki

Ion Androutsopoulos

November 2020

Abstract

In this thesis, we experiment with automatic quality and content evaluation measures for summaries produced from articles of different domains that have been automatically summarized. The ultimate goal is to propose an evaluation measure that can be used by users who want to assess how good a system summary is in terms of quality and content preservation, compared to the original document(s), without the need for human-written references. First, we consider the existing and commonly used metrics for summary evaluation and the limitations they have. Generally, we investigate the evaluation task for summaries generated by different types of models. We develop measures that we evaluate on biomedical and news summaries, we compare them with other state-of-the-art measures and explore their correlations alongside the absolute error with respect to human judgments of summary quality.

Περίληψη

Στην παρούσα διπλωματική εργασία διερευνούμε αυτόματα μέτρα αξιολόγησης της ποιότητας και του περιεχομένου περιλήψεων άρθων από διαφορετικές θεματικές περιοχές, οι οποίες έχουν παραχθεί αυτόματα. Ο στόχος αυτής της μελέτης είναι να προτείνουμε ένα μέτρο αξιολόγησης το οποίο θα μπορεί να χρησιμοποιηθεί από κάθε χρήστη που επιθυμεί να αξιολόγηση την ποιότητα μίας αυτόματης περίληψης, όσον αφορά τόσο στη διατήρηση του περιεχομένου του πρωτότυπου κειμένου, όσο τα ποιτικά και γλωσσικά της χαρακτηριστικά. Επιπρόσθετα, στόχος μας είναι το προτεινόμενο μέτο αξιολόγησης να μπορεί να δώσει ικανοποιητικά αποτελέσματα χωρίς την ανάγκη πρόσβασης σε περιλήψεις που έχουν γραφτεί από ανθρώπους. Αρχικά, αναλύουμε τα υπάρχοντα και ευρέως χρησιμοποιούμενα μέτρα αξιολόγησης και αναλύουμε τις αδυναμίες που μπορεί να παρουσιάζουν. Εξετάζουμε τη διαδικασία αξιολόγησης περιλήψεων σε περιλήψεις που προέρχονται από μοντέλα διαφόρων ειδών. Αναπτύσσουμε μέτρα αξιολόγησης τα οποία εξετάζουμε σε βιοϊατρικά δεδομένα και δεδομένα ειδήσεων, συγκρίνοντάς τα με άλλα κορυφαία (state-of-the-art) μέτρα υπολογίζοντας πόσο απέχουν οι εκτιμήσεις τους τους από τις ανθρώπινες αξιολογήσεις και πόσο καλά μπορούν να συσχετιστούν με αυτές.

Acknowledgements

I would like to express my sincere thanks to my supervisor Ion Androutsopoulos for his constant support and the opportunity he gave me to do research on this interesting field. I would also like to express my gratitude to my co-supervisors, Prodromos Malakasiotis and Marianna Apidianaki for their constant encouragement providing invaluable guidance. They have taught me the methodology to carry out the research and to present the research work as clearly as possible. It was a great privilege and honor to work under their guidance. Many thanks to the members of AUEB's NLP Group for many interesting meetings, motivations and discussions we had during this work. Finally, I couldn't forget to thank my family for their support and my friends for always being there for me.

Contents

1 Introduction

Summarization is the task of identifying and combining the most important information from one or several documents to produce a summary. Over the past few years, neural summarizers (Rush et al., 2015; See et al., 2017) have significantly advanced the state-of-the-art. However, the evaluation procedure is tedious, since it requires great human effort by skilled annotators who assess the quality of the system generated summaries. More importantly, during the development of summarization models, researchers need a fast, accurate estimation of the performance of their models and relying on human experts is impractical. Hence, the need of finding an automatic measure is considered imperative.

The most commonly used automatic evaluation measures for summarization are ROUGE (Lin and Hovy, 2003; Lin, 2004) and BLEU (Papineni et al., 2002) which count overlapping n-grams between a system generated summary and a human-written summary, called reference. Human references are not easy to produce. Focusing on surface similarities, these metrics do not provide good assessments of summary quality. Hence, their estimations are often inaccurate and the metrics do not always correlate well with human judgments (Stent et al., 2005; Callison-Burch et al., 2006). Actually, even when ROUGE correlates well with human quality scores, this does not always reflect the actual quality of the summaries. Intuitively, the higher correlation indicates that the automatic measure ranks the summaries similarly to human judges compared to other summaries, which is useful in competitions since we can distinguish the summarizers producing better or worse quality summaries. However, many versions of ROUGE, especially those based on overlaps of higher order n-grams, tend to produce low scores with small differences in the ratings and despite the fact that they may correlate well with the manual scores, the differences between systems, as reflected in the mean absolute error (MAE), are very small. In such cases, and in order to better estimate the quality of a summary, MAE is a better evaluation measure.

1.1 Contribution of the thesis

We conduct an extensive analysis of the existing evaluation measures considering two dimensions: (i) the quality of the system generated summaries, hereafter described as *quality estimation*; and (ii) the extent to which the generated summaries contain the most important information expressed in the source document(s), hereafter described as *content estimation*. In both cases, our goal is to propose an evaluation measure that does not require human reference summaries and correlates well with human judgments of summary quality. In our experiments, we thus measure the correlation alongside the Mean Absolute Error (MAE) which shows how close the metric's predictions are to the human quality scores. We use the MAE at the summary level and measure the correlation of our proposed metric with human judgments across documents, hereafter described as *document level* evaluation. This shows how well our proposed metric can rank the summaries that have been automatically produced for a specific source document.

1.2 Outline of the thesis

The rest of the thesis is organized as follows:

- Chapter 2 discusses related work and background information.
- Chapter 3 describes the datasets used in this thesis.
- Chapter 4 presents our quality estimation alongside the experiments and the results.
- Chapter 5 presents our content estimation alongside the experiments and the results.
- Chapter 6 concludes and proposes ideas for future work.

Background 2

2.1 Summarization approaches

Extractive vs. Abstractive summarization: There are two main automatic summarization strategies: (i) extractive summarization (Dorr et al., 2003; Nallapati et al., 2017), where salient sequences (e.g., sentences, n-grams) of the source document(s) are selected and copied directly to the summary; and (ii) abstractive summarization (Nye and Nenkova, 2015; See et al., 2017) where the summarizer has to understand the source document(s) and generate the summary from scratch. A hybrid strategy (Hsu et al., 2018) has also been proposed which combines the extractive and abstractive approaches where the salient sequences of the source document(s) should be identified and paraphrased in order to be included in the summary. Both approaches need to identify the most important passages in the source document(s). In the case of abstractive summarization, however, these need also to be understood in order to be rephrased and used in the summary. In an extractive summarization approach, the salient text sequences can simply by concatenated to form the summary. Hence the abstractive strategy is considered more difficult.

Single-document vs. multi-document summarization: Another axis of differentiation for summarization approaches is the number of source document(s) from which the summary is constructed. A coarse distinction that can be established is between single-document summarization (Litvak and Last, 2008; See et al., 2017) and multi-document summarization (Erkan and Radev, 2004; Radev et al., 2004). Single-document summarization aims at finding the key points of a single source document, and generating a summary according to the strategy followed by the summarizer (extractive or abstractive). In multi-document summarization, however, the model needs to extract the most informative points from every source document and identify the ones that should be included and synthesized in the generated summary so as to produce a readable output. A challenge faced by models in both cases, is the redundancy of information in the generated summary. This is less of a problem in single-doc summarization, since ideas might be expressed only once in the source text; but in multi-doc summarization, the same ideas might be expressed several times and with slight variations, in more than one documents. Therefore, in multi-document summarization, the salient sequences should be located in the source documents and appropriately filtered in order to avoid redundant information in the summary. This constitutes an additional difficulty in case of multi-document summarization, compared to single-document.

2.2 Summary evaluation approaches

Content-based vs. Quality-based evaluation: Automatically produced summaries can be evaluated extrinsically, as to whether they can serve in specific downstream tasks (for example, question answering), or intrinsically (Steinberger and Jezek, 2012). *Extrinsic* measures assess whether the automatically produced summaries can help in a given task; in the case of question answering, they estimate how useful the summary can be for answering specific questions. On the other hand, *intrinsic* methods focus on the quality and content of the automatically generated summary. This is the kind of evaluation we address in this work.

The intrinsic evaluation measures can be further distinguished into *quality* based and *content* based. Quality evaluation measures analyze the quality characteristics of the system generated summary without need to access other documents. We use the term "quality characteristics" to refer to the properties that we want, ideally, the summary to have, such as readability, fluency, non-redundancy, etc. All of the above mentioned criteria can be examined by only accessing the system-generated summary. On the contrary, the content-based evaluation measures, in most cases, involve a comparison of the summary to another document(s), which serves to assess whether the summary captures the key points of content. The document(s) used for the comparison can be either the source document(s), or some human-written summaries (called references) which include the important information from the source document(s).

Reference-based vs. reference-free measures: This distinction applies to content-based evaluation metrics, which involve a comparison of the summary to other documents. We describe as "reference free" the metrics that do not use any human-crafted summary (reference), and only rely on the source documents(s). We describe as "reference-based" the metrics that use human-written references for comparison. These reference texts are generally produced by human annotators who are asked to read the original document(s) and produce shorter texts carrying the same information.

Supervised-based vs. Non-supervised measures: Human evaluation of automatic summarization output is an expensive procedure, there, however, exist supervised measures that leverage available human annotations. These try to evaluate system-generated summaries in a way similar to that used by human annotators to evaluate their quality. Supervised metrics are trained on "gold" ground truth assessment scores, contrary to unsupervised metric. These measures can also be distinguished into quality-based and content-based, depending on whether they evaluate the linguistic qualities of a summary (e.g., grammaticality) or its content (i.e. whether the summary contains the important information from the original document or a reference summary). Also, the content-based ones, can be either reference-free or reference-based, as described above.

2.3 Assessing the quality of summarization evaluation metrics

Evaluation measures can also be compared and evaluated, in respect to how well they approximate human quality assessments. A straightforward way to do so is to compare the estimations produced by an automatic metric for a summary to the assigned manual scores. The comparison can involve measuring the correlation (e.g., Spearman ρ, Kendall τ, Pearson r) of the two scores, or calculating the mean absolute error (MAE) between them.

Measuring correlation and error: The most common way to compare the predictions made by an automatic metric with the human judgments is to assess the performance of automatic measures using correlation measures. The most commonly used correlation measures are Kendall τ (Puka, 2011), Spearman ρ (Sedgwick, 2014) and Pearson r (Kirch, 2008). Kendall's correlation is a non-parametric statistic dependence test based on τ-coefficient and takes values in the interval [-1, 1]. Intuitively, the higher values indicate a stronger association of the automatic and the manual ranking compared to lower values. A zero correlation indicates the dissimilarity of the two rankings. Let n be the number of summaries for which a human score $h_1, ...h_n$ and an automatically assigned score $a_1, ...a_n$ are available. The pair (h_i, a_i) corresponds to the human and the automatic scores assigned to a summary i. In the case of system-level evaluation, an observation corresponds to a system (rather than a summary). A pair of observations (h_i, a_i) and (h_j, a_j), where $i < j$, is said to be concordant if $h_i < h_j$ and $a_i < a_j$ or $h_i > h_j$ and $a_i > a_j$; otherwise the pair is said to be discordant. When $h_i = h_j$ and $a_i = a_j$, the pair is neither concordant nor discordant. Also, let C_{con} and C_{dis} be the number of concordant and discordant pairs of observations respectively. When all the human scores are equal to the manual ones, the correlation cannot be calculated. In all the other cases, the Kendall's τ correlation can be calculated as:

$$\tau = \frac{C_{cor} - C_{dis}}{\binom{n}{2}} = \frac{C_{cor} - C_{dis}}{\frac{n(n-1)}{2}} = \frac{2(C_{cor} - C_{dis})}{n(n-1)} \tag{2.1}$$

Spearman's ρ coefficient is similar to Kendall's τ. It is a also non-parametric measure of rank correlation and takes values in the interval [-1, 1]. Specifically, it assesses the monotonic relationships of two distributions. It is equal to Pearson's correlation, which is described below, between the rank values of those two distributions. Let $R_h = R_{h_i}, ..., R_{h_n}$ and $R_a = R_{a_i}, ..., R_{a_n}$ be the rank values of the distributions h and a respectively. The observation (R_{h_i}, R_{a_i}) corresponds to the rank of h_i in the human scores distribution and the rank of a_i in the corresponding automatic scores distribution described above (e.g., $(1, 5)$). Let n be the number of summaries; 1 denotes the first position in the ranking (not the lowest value) in the corresponding distribution and n the highest (position in the ranking). The Spearman's ρ correlation can be calculated as:

$$\rho = \frac{Cov(R_h, R_a)}{\sigma_{R_h} \sigma_{R_a}} \tag{2.2}$$

$Cov(R_h, R_a)$ are the covariances and σ_{R_h}, σ_{R_a} the standard deviations of R_h and R_a distributions respectively. Similarly to Kendall's correlation, Spearman's correlation takes higher values when there is a strong association between the automatic and the manual ranking and lower otherwise. While Spearman's correlation assesses monotonic relationships of the compared distribution, the last correlation measure, Pearson r, assesses the linear relationships of the compared distributions and can be calculated as:

$$r = \frac{\sum_{i=1}^{n}(h_i - \bar{h})(a_i - \bar{a}))}{\sqrt{\sum_{i=1}^{n}(h_i - \bar{h})^2} \sqrt{\sum_{i=1}^{n}(a_i - \bar{a})^2}} \tag{2.3}$$

where $\bar{h}$ and $\bar{a}$ are the arithmetic means of the human and automatic score distributions, respectively.

Finally, two distributions can also be compared by the absolute error which shows how far are, on average, the estimations produced by an automatic measure from the human scores. We used in our experiments the mean absolute error (MAE) which is calculated as follows:

$$\text{MAE} = \frac{\sum_{i=1}^{n} |h_i - a_i|}{n} \tag{2.4}$$

Also, in our results we present an average of the above measures across multiple evaluations alongside the standard error of the mean (SEM) which shows how far a sample mean is likely to be from the actual mean of the whole distribution. SEM can be calculated as:

$$\text{SEM} = \frac{\sigma}{\sqrt{c}} \tag{2.5}$$

Where σ is the standard error of the sample and c the number of the random samples. The adequacy of a specific metric for an evaluation, depends on the evaluation level on which it will be applied.

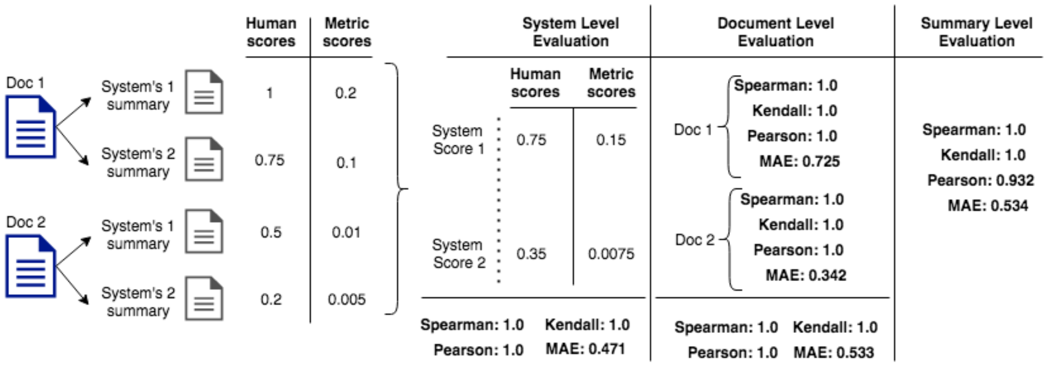

Figure 2.1.: An example showing the three evaluation levels.

Evaluation at different levels: An automatic metric can be evaluated at the summary, document or the system level. The simplest way to compare an automatic to a human-measure is the "summary" level. At this level, the summaries predictions of the automatic metric are directly compared to the quality scores assigned by the human judges. In spite of its simplicity, this comparison might not always lead to safe conclusions. The reason is that in the summary-level evaluation, automatic predictions and human scores of summaries from different source documents are mixed. Some documents might be easier to summarize than others; as a consequence, their summaries are generally ranked higher by the human judges and the automatic metrics. This might lead to inaccurate conclusions with a correlation metric that accounts for the exact order of the compared distributions.

On the other side, Mean Absolute Error (MAE) can provide an indication of the distance between automatic and human scores. We can see in Figure 2.1 that the correlation measures at the "summary" level may produce scores equal to 1, indicating that the automatic measure is very good and managed to rank the summaries with the same order as the one assigned by the annotators. However, these measures are bad for assessing the actual quality of the summaries, because if we compare, one by one, the scores assigned by the judges with the corresponding scores produced by the automatic metric, the absolute error between each pair is too big. An evaluation at the "document" level, using either the correlation measures or MAE, can serve to remedy the problem of the mixed scores occurred at the summary level evaluation. In this case, the automatic predictions and the human scores assigned to the summaries, are separated according to the source document they summarize. After this separation, we can compare the corresponding distributions per document, coming up with a correlation and a MAE score. In order to extract on overall score for each measure (correlation-based or MAE), for the entire dataset, we can aggregate by averaging all those scores that correspond to each document, as in Figure 2.1. An evaluation at the "system level" is suitable for challenges where the goal is to rank summarization systems according to their performance. In this case, we aggregate the human scores assigned to the summaries, and the automatic predictions, according to the system that produced them. By averaging the human and the measurement's scores we can obtain a human score and a measurement's score per system as in Figure 2.1. At the system level, we evaluate the ranking of the systems by the automatic metric and the human judges. Hence, the correlation based measurements which takes into account the exact order of the compared distributions, are preferred than the MAE at the system level evaluation. We can see in Figure 2.1 that despite the big MAE which indicates that the estimations are too far from the human scores, the automatic measure managed to rank the systems in the same order as the judges and achieved correlation equal to 1. This indicates that, in this specific level, the final ranking of the systems would be the same if we reported the automatic or the human ranking. In the experiments of this work, we report either the MAE or the correlations at the document level. We put special focus to the MAE metric because it indicates how better the automatic evaluation measures assess the quality of the summary than the correlation metrics.

2.4 Limitations of existing summarization evaluation methods:

The most commonly used evaluation metrics in summarization are ROUGE (Lin and Hovy, 2003; Lin, 2004) and BLEU (Papineni et al., 2002). BLEU is widely used in machine translation (MT) but both measures can be also used to evaluate a system-generated summary. Despite their popularity, many works (Stent et al., 2005; Callison-Burch et al., 2006) have shown that BLEU and similar measures based on n-gram overlap do not correlate well with human judgements. ROUGE and BLEU rely on n-gram overlaps between the system-generated text and the human-reference(s). Many ROUGE and BLEU versions are available but it is not trivial to decide which one is the best to use (Graham, 2015). The BLEU metric exists in three variations. In the first one, each n-gram in the reference summary is matched at most once to an n-gram from the system-generated text. In the second one, each n-gram in the reference summary can be matched multiple times to the common n-grams dividing the total matches with the total number of n-grams in the system-generated text. Finally, the third version includes a brevity penalty to discourage the matches of very short sequences. However, the most popular variation calculates the BLEU scores (of the first version) for multiple values of n (e.g. n = 1, 2, 3, 4) averaging them to produce the final score.

The simplest versions of ROUGE are inspired by BLEU. These are ROUGE -1 which considers unigrams (i.e., words) and ROUGE -2, -3, -4 which consider bigrams, trigrams, and 4-grams respectively. There are also the ROUGE-L and ROUGE-W versions which are based on the longest common subsequence (LCS) and the weighted LCS statistics, respectively. In addition, the ROUGE-S4 and ROUGE-SU4 rely on skip-gram based co-occurrence statistics, with the difference that ROUGE-SU4 considers also unigram overlaps. We should note that the above described ROUGE versions, can be calculated with respect to the length of the reference (e.g., ROUGE-L -RECALL), or the system summary (e.g., ROUGE-L -PRECISION) or the combination of them (e.g., ROUGE-L -F1). In all the ROUGE measures, it can be applied stop-words removal and stemming in order to obtain more matches. ROUGE and BLEU, as recall-oriented measures, can be used also to assess how well a system summary preserves the content of the original text by comparing it to the human reference, hence they are categorized to the content-based measures.

Another n-gram based evaluation metric, which is mostly used in the machine translations tasks, is METEOR. METEOR relies on unigram overlap between system-generated text and the human-produced references while several improvements have been published such as METEOR 1.5 (Denkowski and Lavie, 2014) which weighs content and function words differently by varying the importance assigned to different types of matching (e.g., exact matching, stemmed, etc). Afterward, Guo and Hu (2019) proposed METEOR 2++ which further incorporates a learned external paraphrase resource. A popular measure in summarization evaluation is also PYRAMID (Nenkova and Passonneau, 2004). PYRAMID relies on information units expressed in the candidate summary and reference which are called symmarization content units (SCUs). The more times a SCU is found in the human-written summaries, the bigger weight it will have. Thus, a pyramid is created, where the most important SCUs are placed at the top. Therefore, if a system-generated summary contains more SCUs that are placed at the top of the Pyramid, the higher PYRAMID score will be assigned to it. Therefore, all the above measures are reference based. However, there is a measure called HIGHRES (Hardy et al., 2019) which is also content-based as BLEU, but relies on highlighted snippets, which are extracted, by annotators, from the source document and capture the salient content. The difference between BLEU and METEOR is that BLEU is reference-based and HIGHRES is reference-free. Therefore, both measures need an expert either to annotate the salient sequences on the source document (HIGHRES) or to develop summaries that capture the salient content in order to be used on the evaluation process (BLEU).

Apart from the n-gram based measures, there are embedding based measures such as MEANT2.0 (Lo, 2017) and YISI-1 which use word embeddings and shallow semantic parsing to compute structural and lexical similarities.[1] These embedding-based measures inspired Zhang et al. (2020) who proposed BERT SCORE. BERT SCORE relies on the BERT language representation model (Devlin et al., 2019) and computes a similarity score for each token in the candidate sentence with each token in the reference sentence. For the calculation of the similarity it uses the contextual embeddings produced by a SENTENCE-BERT encoder (Reimers and Gurevych, 2019) which is a modification of the pretrained BERT model and can capture paraphrases and longer dependencies on the text than the classic n-gram based measures. Also, Gao et al. (2020) proposed SUPERT which is similar to BERT SCORE with the difference that SUPERT calculates the similarity of the candidate summary with a pseudo-reference constructed by the proposed mechanism concatenating the first 10 sentences of each source document. However, these measures use the BERT model without

[1] In YISI-1 semantic parsing is optional.

any further training, relying on the representations extracted from the frozen BERT. BLUERT, proposed by Sellam et al. (2020), and Sum-QE proposed by Xenouleas et al. (2019), train a BERT model to learn better representations, capturing more information about the content of each word, which can help the quality estimations. We should note that these two methods (BLUERT and Sum-QE) are trained to score quality aspects of the summaries (e.g., grammaticality, fluency, etc.) which indicates that the contextual embeddings can be used not only for content evaluation but can help at the quality estimation, too. Specifically, the Sum-QE is constituted by a BERT encoder, which converts the summary to a dense vector representation, and a linear regressor (LR) which learn to predict a quality score. The model was trained using two settings, the single task (ST) and the multi task learning (MT). On the single task learning, the model uses one BERT encoder and a linear regressor per quality score that tries to learn predict. On the multi task setting, one BERT encoder is used and as many linear regressors as the number of the different quality scores that tries to learn predict. However, the performance of these metrics, which require training, can be compromised when moving to new domains, even within a known language and task (Chaganty et al., 2018). In our work, we evaluate the metrics on summaries across different domains in order to have a better knowledge about their performance.

3 Datasets

In this chapter we discuss the three datasets used in this thesis. The first two datasets, DUC and NEWSROOM, contain articles and summaries from the news domain while the third one comes from a challenge on large-scale biomedical semantic indexing and question answering (QA), called BioASQ.[1][2][3]

3.1 DUC

We use the datasets from the NIST DUC-05 (Dang, 2006a), DUC-06 (Dang, 2006b) and DUC-07 (Over et al., 2007) shared tasks. The tasks were similar at each year: given a question and a number of relevant news-wire articles the contestants were asked to synthesize a fluent, well-organized summary. Each submitted summary should answer the question and contain up to 250 words. Every question has been answered by the judges, so there are available also some human-written summaries which can be used in the evaluation process in order to be compared with the system-generated summaries. DUC-05 contains 1,600 summaries (50 questions x 32 systems); in DUC-06, 1,750 summaries are included (50 questions x 35 systems); and DUC-07 has 1,440 summaries (45 questions x 32 systems). Each submitted summary was evaluated according to the following five linguistic quality aspects and the guidelines were given for each one are:

- $\mathcal{Q}$**1 – Grammaticality:** The summary should have no datelines, system-internal formatting, capitalization errors or obviously ungrammatical sentences (e.g., fragments, missing components) that make the text difficult to read.
- $\mathcal{Q}$**2 – Non redundancy:** There should be no unnecessary repetition in the summary.
- $\mathcal{Q}$**3 – Referential Clarity:** It should be easy to identify who or what the pronouns and noun phrases in the summary are referring to.
- $\mathcal{Q}$**4 – Focus:** The summary should have a focus; sentences should only contain information that is related to the rest of the summary.
- $\mathcal{Q}$**5 – Structure & Coherence:** The summary should be well-structured and well-organized. It should not be just a heap of related information, but it should constitute from sentence to sentence a coherent body of information about a topic.

[1] See https://duc.nist.gov/data.html
[2] See http://lil.nlp.cornell.edu/newsroom/
[3] See http://bioasq.org/

3.2 Newsroom

The NEWSROOM dataset (Grusky et al., 2018) contains approx. 1.3 million news articles, along with summaries written by the authors of the articles or editors, and was developed to train summarization systems. The articles cover a wide range of topics (general news, sports, entertainment, financial) and the (human-authored) summaries use many different summarization styles. Several baseline and state of the art summarizers have been trained and tested on Newsroom. Baselines include:

- The Lede-3 (Nallapati et al., 2017) which copies the first sentence, first paragraph, or first n words of the source text.
- The Extractive Oracle Fragments (Grusky et al., 2018) an oracle that examines the reference (gold, human-authored) summary, and produces a concatenation of the longest shared token sequences from the article in the order they appear in the reference summary
- The TextRank (Mihalcea and Tarau, 2004) an unsupervised sentence-ranking approach which uses the PageRank algorithm (Page et al., 1999), over a graph which represents the article's sentences as nodes, collecting the most significant of them in the order they appear in the article.

State of the art summarizers include:

- The abstractive summarizer of Rush et al. (2015), who use a sequence-to-sequence model with attention.
- The Pointer networks (See et al., 2017) which combine extractive and abstractive summarization strategies.

In their work, Grusky et al. (2018) asked human annotators to score in total 420 system generated summaries. More precisely, 7 of the above mentioned systems (Abstractive, Fragments, Lede3, Pointer_c, Pointer_n, Pointer_s and Textrank) constructed a summary for each one of the 60 randomly selected articles from the released test set. Pointer_c, Pointer_n and Pointer_s are the Pointer networks with the difference that the first one is trained on the CNN / Daily mail dataset (Hermann et al., 2015), the second one on the Newsroom dataset, and the last one on a random subset of Newsroom training data but with a size equal to the CNN / Daily Mail training set. Each summary has been evaluated according to four dimensions:

- **Coherence**: Do phrases and sentences of the summary fit together and make sense collectively?
- **Fluency**: Are the individual sentences of the summary well-written and grammatically correct?

- **Informativeness**: How well does the summary capture the key points of the article?
- **Relevance**: Are the details provided by the summary consistent with the details in the article?

Also, Grusky et al. (2018) provide three more scores that can be computed automatically for each summary given only the source article.

Coverage intuitively measures the percentage of summary words that come from the original article. Below S and A are the summary and article, respectively, $|\cdot|$ denotes the length in words, and $F(A, S)$ is the set of the longest shared word sequences between the summary and the article.

$$\mathit{Coverage}(A, S) = \frac{1}{|S|} \sum_{f \in F(A,S)} |f| \tag{3.1}$$

Density rewards summaries consisting of longer fragments of the article.

$$\mathit{Density}(A, S) = \frac{1}{|S|} \sum_{f \in F(A,S)} |f|^2 \tag{3.2}$$

Compression measures the ratio of the article length to the summary length.

$$\mathit{Compression}(A, S) = \frac{|A|}{|S|} \tag{3.3}$$

3.3 BioASQ

BioASQ (Tsatsaronis et al., 2015) is a challenge on biomedical retrieval question answering (QA). Given a question, the participants are required to provide an answer in the required format depending on the question type (e.g., yes/no, factoid, list) alongside a summary that supports the answer. The summaries are produced by the participating systems using the provided relevance articles and the marked snippets (e.g., sentences) annotated by the experts. Also, it should be mentioned that each question of the dataset has been answered by at least one expert so there is at least one human written summary that can be used for the evaluation process and can be compared with the system-generated summaries. We used the data from 6 years of the challenge (2014–2019). We split the data into training, validation and test sets by taking the years 2014–2017 for training, 2018 for validation and 2019 for testing. The training data were used only when the evaluation measure required supervision. The evaluation was conducted using the test data, while the validation data was used for tuning purposes.

For the evaluation process, each summary has been evaluated according to following dimensions:

- ***Information Recall***: The extent to which the summary contains all the necessary information.
- ***Information Precision***: The extent to which the summary does not contain irrelevant information.
- ***Repetition***: The extent to which the summary does not contain the same information more than once.
- ***Readability***: The extent to which the summary is readable.

These dimensions and further guidelines were given by the BioASQ organizers who provide a reference (expert's summary) to the appropriate BioASQ deliverable about evaluation. We observe that the *Informativeness* and *Relevance* dimensions of the NEWSROOM dataset are defined similarly to *Information Recall* and *Information Precision* in the BioASQ dataset respectively, which are analyzed below (Section 5) in more details.

4 Quality estimation for text summarization

4.1 Introduction

Quality Estimation (QE) is well established in MT (Bojar et al., 2016; Bojar et al., 2017). QE methods provide a quality indicator for translations at run-time without relying on human references, typically needed by MT evaluation measures (Papineni et al., 2002; Banerjee and Lavie, 2005; Denkowski and Lavie, 2014). QE models for MT make use of large post-edited datasets, that contain machine-generated translations, and apply machine learning methods to predict post-editing effort scores and quality (good/bad) labels. We apply QE to summarization focusing on quality aspects that reflect the coherency, the repetition, the fluency and the readability of the generated texts. Specifically, we try to predict the quality-based scores provided along with the summaries of the corresponding datasets (Section 3). We refer to the *Coherence* and *Fluency* scores of NEWSROOM dataset and the *Repetition* and *Readability* scores of the BioASQ dataset.

4.2 Methods

In this Section, we discuss the methods used for quality estimation. We split them into two categories, the methods used to assess the quality of the summaries from the news domain using the NEWSROOM dataset and the methods used to assess the quality of the summaries from the biomedical domain using the BioASQ dataset.

4.2.1 News domain

In this Section, we focus on the NEWSROOM dataset and we try to find (or develop) methods that can estimate the *Fluency* and the *Coherence* human quality scores that are provided alongside the summaries.

Density: *Density*, as mentioned in Section 3.2 is an automatically produced measure published by Grusky et al. (2018).[1] Intuitively, the more sequences copied from the source

[1] See the Equation 3.2.

document(s) to the summary, the more fluent and coherent it would be, since these sequences are well written and structured by the author(s) of the source document(s). Hence, we expect that higher scores of *Density* should correspond to higher scores on *Fluency* and *Coherence* too. We compared the *Density* scores with the manual scores of *Fluency* and *Coherence* and the results can be found in Table 4.1.

Coverage Similarly to *Density*, *Coverage* is also an automatically produced measure published by Grusky et al. (2018). Intuitively, *Coverage* measures the percentage of the summary words that come from the source document so we expect a similar behavior to *Density*, higher scores of *Coverage* should correspond to higher scores on *Fluency* and *Coherence*, too. We compared the *Coverage* scores with the manual scores of *Fluency* and *Coherence* and the results can be found in Table 4.1.

SUM-QE $\mathcal{Q}1$ (ST) & SUM-QE (MT) $\mathcal{Q}1$: In our experiments, we wanted to include the existing predictors of SUM-QE trained on the DUC data as in our previous work Xenouleas et al. (2019). However, in the previous work, we trained the SUM-QE model using a leave-one-year-out procedure in order to be able to evaluate it in all of the three DUC datasets (2005–2007) separately. In this work, since we evaluate on the NEWSROOM dataset, we trained the SUM-QE model using all the three DUC datasets, utilizing all the available data. Following the training methods presented in our previous work, we trained the SUM-QE model using the single-task (ST) and multi-task (MT) learning settings. In single-task learning, we use a separate estimator, one per quality score ($\mathcal{Q}1$, $\mathcal{Q}5$), each having its own encoder (BERT instance) and a linear regression layer on the top in order to learn predict the corresponding quality score. In multi-task learning, the SUM-QE model uses multiple linear regression layers on the top of the same BERT instance in order to learn predict more than one quality scores simultaneously. The main difference in this work is that we use only the $\mathcal{Q}1$ (*Grammaticality*) and $\mathcal{Q}5$ (*Structure & Coherence*) scores in the multi-task learning setting which are semantically closer to *Fluency* and *Coherence* respectively avoiding to mix irrelevant quality aspects that may harm our predictions. Intuitively, the more structured and grammatically correct the summary is, the more well written and fluently it would be. Hence, higher scores of *Grammaticality* ($\mathcal{Q}1$) and *Structure & Coherence* ($\mathcal{Q}5$) should correspond to higher scores of *Fluency*. Therefore, SUM-QE $\mathcal{Q}1$ (ST) corresponds to the $\mathcal{Q}1$ estimations of the pretrained SUM-QE model on the DUC data using only the $\mathcal{Q}1$ scores to drive the training while SUM-QE (MT) $\mathcal{Q}1$ corresponds also to $\mathcal{Q}1$ estimations of the pre-trained SUM-QE model on DUC data, using though both the $\mathcal{Q}1$ and $\mathcal{Q}5$ scores to drive the training. These two mentioned measures were compared only with the *Fluency* scores and the results can be found in Table 4.1.

SUM-QE $\mathcal{Q}5$ (ST) & SUM-QE (MT) $\mathcal{Q}5$: These two methods are similar to the previous ones. Similarly, we trained the SUM-QE model using the single-task (ST) and multi-task (MT) learning. We only used the $\mathcal{Q}1$ (*Grammaticality*) and $\mathcal{Q}5$ (*Structure & Coherence*) scores in the multi-task learning since the more structured and grammatically correct is a summary, the more coherent it would be. Hence, higher scores of *Grammaticality* ($\mathcal{Q}1$) and *Structure & Coherence* ($\mathcal{Q}5$) should correspond to higher scores of *Coherence*. Therefore, SUM-QE $\mathcal{Q}5$ (ST) corresponds to the $\mathcal{Q}5$ estimations of the pre-trained SUM-QE model on the DUC data, using only the $\mathcal{Q}5$ scores to drive the training and SUM-QE (MT) $\mathcal{Q}5$ corresponds also to $\mathcal{Q}5$ estimations of the pre-trained SUM-QE model, on the DUC data, using though both the $\mathcal{Q}1$ and $\mathcal{Q}5$ scores to drive the training. These two mentioned measures were compared only with the *Coherence* scores and the results can be found in Table 4.1.

AVG: Since the above described methods seemed to work well, we decided to combine them using a simple way at first. For *Coherence* estimation, we averaged the scores from the frozen $\mathcal{Q}5$ predictor of SUM-QE, trained on the DUC data, with *Coverage* and *Density* scores. A similar procedure was applied to the *Fluency* estimation we averaged the scores from the frozen $\mathcal{Q}1$ predictor of SUM-QE, trained on the DUC data, with *Coverage* and *Density*. The results can be found in the Table 4.1 as AVG ($\mathcal{Q}1$, DENSITY, COVERAGE) on *Fluency* estimation and AVG ($\mathcal{Q}5$, DENSITY, COVERAGE) on *Coherence* estimation.

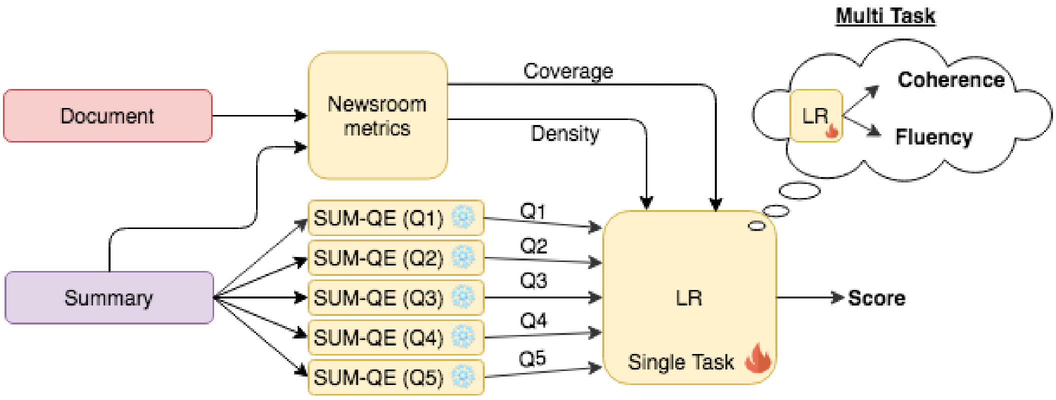

Figure 4.1.: Training process of the linear regressor using the frozen SUM-QE predictors, trained on the DUC datasets using single-task learning, and the automatically produced scores *Coverage* and *Density* that provided along the NEWSROOM dataset.

Linear regression (LR): We did not just rely only on the simple combination using average, we also used a linear regressor (LR) to learn to weight appropriately the linguistic quality aspects alongside the *Density* and *Coverage* scores in order to produce better quality estimations. Since we have ensured that, using the LR, the input scores will not be equally treated, as in the above case where the average was used, we decided to include the predictions from all the pre-trained, on DUC data, SUM-QE predictors ($\mathcal{Q}1$,...,$\mathcal{Q}5$) to the model. Hence, the estimations from the SUM-QE predictors alongside the *Density* and the *Coverage* scores, form the input of the model. We trained the model using single-task and multi-task learning, as described above. In Figure 4.1, we can see an illustration of the pipeline. Since he had only 420 human evaluated system summaries, as mentioned in Section 3.2, we trained the LR using a 5-fold cross validation procedure. For each fold, the model was trained six independent times. The predictions for each test summary of the fold, were calculated by averaging the six predicted scores by each independent training procedure.[2] The results can be found in Table 4.1 as LR ($\mathcal{Q}1, \ldots, \mathcal{Q}5$, DENSITY, COVERAGE). Additionally, we checked how well can estimate the *Fluency* and the *Coherence* scores an average of the predictions from all the SUM-QE predictors ($\mathcal{Q}1$,...,$\mathcal{Q}5$) alongside the *Density* and the *Coherence* and can be found as AVG ($\mathcal{Q}1, \ldots, \mathcal{Q}5$, DENSITY, COVERAGE) in Table 4.1.

4.2.2 Biomedical domain

In this Section, we focus on the BIOASQ dataset and try to find (or develop) methods that can assess the quality of a biomedical summary observing the amount of repetition in the text and how readable it is. We try to estimate the *Repetition* and *Readability* scores provided alongside the summaries of the BIOASQ dataset.

SUM-QE $\mathcal{Q}1$ (ST) & SUM-QE $\mathcal{Q}2$ (ST): Similarly to the news data, we used the pre-trained, on the DUC dataset, SUM-QE predictors $\mathcal{Q}1$ (*Grammaticality*) and $\mathcal{Q}2$ (*Non redundancy*) using single-task learning (ST) in order to estimate the *Readability* and the *Repetition* of a biomedical summary respectively. Intuitively, a grammatically correct summary will be easier to be read, so the higher scores of *Grammaticality* should correspond to higher scores of *Readability*. Similarly, the less redundant information a summary contains, the fewer repetitions it will have, so the higher scores of *Non redundancy* should correspond to higher scores of *Repetition*, according to the exact definitions (Section 3). Hence, we compared the estimations of the *Grammaticality* with the *Readability* scores and the estimations of *Non redundancy* with the *Repetition* scores. The results are shown in the Table 4.2 as SUM-QE $\mathcal{Q}1$ (ST) and SUM-QE $\mathcal{Q}2$ (ST).

SUM-QE READ (ST), SUM-QE (MT) READ & SUM-QE REP (ST), SUM-QE (MT) REP: Similarly to the methods SUM-QE $\mathcal{Q}1$ (ST), SUM-QE (MT) $\mathcal{Q}1$, we used the SUM-QE model and trained it on the biomedical summaries. The difference is that we used the *Readability* and the *Repetition* scores on the single-task or a multi-task learning in order to drive the training. The results, evaluating on the test data are shown in the Table 4.2.

[2] In the Appendix A.1.1, we can see the MAE curves for each fold.

4.3 Experimental Results

To evaluate our methods for a particular quality-based measure, we used:

1. The micro-average of MAE (Equation 2.4) comparing all the predictions to the human scores of the summaries, regardless of the document (or the question) they summarise (or answer). This is also our main evaluation factor and the comparison between each measurement was conducted by comparing the MAE scores achieved from the human scores.
2. The macro-average of MAE (Equation 2.4) across documents (or questions).[3] Alongside the macro-average of MAE, we present the standard error of the mean (SEM) after the ± symbol (Equation 2.5).
3. The Spearman's ρ (Equation 2.2), Kendall's τ (Equation 2.1) and Pearson's r (Equation 2.3) correlations at the Document level (Section 2.3). Alongside with the correlations, we also present the standard error of the mean (SEM) after the ± symbol (Equation 2.5).

We should mention that the *Density* scores, produced using the Equation 3.2, and the Sum-QE scores weren't in the range [0,1] like the manual scores, so the Mean Absolute Error (MAE) could not be calculated correctly. For this reason, we normalized the predicted scores produced by these two methods using the formula of z-transformation (4.1) and a 5-fold cross-validation procedure to compute the MAE. In each fold, the scores from one set were normalized using μ and σ calculated on the four remaining sets. The *Coverage* scores produced by the Equation 3.1 and all the other methods were already in [0,1] so no further processing was required.

$$Z\_score = \frac{\overline{x} - \mu}{3\sigma} \tag{4.1}$$

Since 3σ in the denominator of the equation captures the 99.9% of a normal distribution, we clipped the Z_score to [0,1] to ensure that all the predictions would be in this range. We also kept the original scores to calculate the correlations, since the scores do not need to be normalized in this case.

Before analyzing the results from the models that required training, we should notice, in Table 4.1 the *Density* measure provides a very good estimation for both the *Coherence* and *Fluency* scores achieving very small MAE. On the other hand, the Sum-QE predictors, trained on the DUC data, did not reduce as much the MAE as we expected. On the *Fluency* estimation, a small reduction was obtained, from 0.166 (that *Density* achieved) to 0.161

[3] On NEWSROOM dataset, all the 7 systems constructed a summary for all the 60 articles, so none is missing. Therefore, the MAE on the micro is the same with MAE on the macro level.

		NEWSROOM quality estimations			
	Method	MAE	ρ	τ	r
Coherence	*Coverage*	0.300	0.379 ± 0.045	0.326 ± 0.037	0.522 ± 0.266
	Density	0.155	0.675 ± 0.030	0.574 ± 0.574	0.656 ± 0.220
	SUM-QE Q5 (ST)	0.163	0.519 ± 0.046	0.437 ± 0.040	0.533 ± 0.045
	SUM-QE (MT) Q5	0.175	0.398 ± 0.053	0.339 ± 0.045	0.403 ± 0.055
	AVG (Q5, Density, Coverage)	0.139	**0.692 ± 0.033**	**0.588 ± 0.031**	**0.699 ± 0.031**
	AVG (Q1, ..., Q5, Density, Coverage)	0.145	0.610 ± 0.039	0.518 ± 0.036	0.636 ± 0.038
	LR (Q1, ..., Q5, Density, Coverage)	**0.126**	0.514 ± 0.045	0.452 ± 0.04	0.568 ± 0.040
Fluency	*Coverage*	0.299	0.334 ± 0.047	0.285 ± 0.040	0.460 ± 0.347
	Density	0.166	0.636 ± 0.032	0.533 ± 0.030	0.625 ± 0.209
	SUM-QE Q1 (ST)	0.161	0.587 ± 0.040	0.488 ± 0.036	0.597 ± 0.037
	SUM-QE (MT) Q1	0.168	0.544 ± 0.035	0.432 ± 0.032	0.553 ± 0.035
	AVG (Q1, Density, Coverage)	0.151	**0.665 ± 0.031**	**0.557 ± 0.029**	**0.670 ± 0.032**
	AVG (Q1, ..., Q5, Density, Coverage)	0.152	0.624 ± 0.036	0.517 ± 0.033	0.660 ± 0.035
	LR (Q1, ..., Q5, Density, Coverage)	**0.121**	0.572 ± 0.037	0.459 ± 0.033	0.608 ± 0.036

Table 4.1.: Mean Absolute Error MAE alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (SEM) between the human (*Coherence* and *Fluency*– Section 3.2) and automatic measures (*Coverage*, *Density*, Sum-QE and combination of Sum-QE predictors) at the document level. Our main evaluation factor in the MAE.

using the Q1 predictor but at the *Coherence* estimation there was not any reduction on the MAE. However, when we combined the *Density* and the *Coverage* scores with the Sum-QE predictors using average, the error dropped in both *Coherence* and *Fluency* measures even when the predictions from one predictor (Q1 or Q5) or from all the predictors (Q1–Q5) were included. The best estimation to *Coherence* and *Fluency* was achieved when we combined all the linguistic quality aspects with the *Density* and *Coverage* using a linear regressor and the MAE dropped to approx. 0.120 from the gold scores. Hence, all the linguistic qualities alongside the *Density* and the *Coherence* scores can help estimate the *Coherence* and the *Fluency* of a generated text.

		BioASQ (2019 Test) quality estimations				
		Micro	*Macro*			
	Method	MAE	MAE	ρ	τ	r
READ	SUM-QE Q1 (ST)	0.263	0.257 ± 0.007	0.233 ± 0.031	0.215 ± 0.029	0.275 ± 0.032
	SUM-QE READ (ST)	**0.190**	**0.187 ± 0.005**	**0.379 ± 0.028**	**0.351 ± 0.026**	**0.409 ± 0.031**
	SUM-QE (MT) READ	0.194	0.189 ± 0.005	0.371 ± 0.028	0.344 ± 0.026	**0.409 ± 0.030**
REP	SUM-QE Q2 (ST)	0.180	0.177 ± 0.004	0.482 ± 0.029	0.443 ± 0.027	0.482 ± 0.030
	SUM-QE REP (ST)	**0.150**	**0.149 ± 0.004**	0.572 ± 0.026	0.532 ± 0.024	0.584 ± 0.026
	SUM-QE (MT) REP	0.165	0.165 ± 0.004	**0.613 ± 0.024**	**0.571 ± 0.023**	**0.614 ± 0.025**

Table 4.2.: Mean Absolute Error MAE on the *micro* level and *macro* MAE alongside the Spearman's ρ, Kendall's τ, Pearson's r correlations ± Standard Error of the Mean (SEM) between human (READ: *Readability*, REP: *Repetition*– Section 3.3) and automatic measure (Sum-QE trained on DUC and BioASQ datasets) at the document level using the test data. Our main evaluation factor in the MAE.

On the biomedical quality estimation, we evaluated our methods using both the test and validation datasets.[4] It should be mentioned that the human scores were in the range [1, 5] and we normalized them in the range [0, 1] to conduct an accurate evaluation. Our

[4] The results of the validation dataset can be found in Table A.1 in the Appendix Section A.1.2.

observations here are that the frozen predictors of the Sum-QE model, trained on news summaries (SUM-QE $\mathcal{Q}1$ (ST), SUM-QE $\mathcal{Q}2$ (ST)), didn't perform so well for estimating the *Readability* and the *Repetition* of biomedical summaries. On the contrary, when we trained the same model (Sum-QE) using as gold scores the *Readability* and the *Repetition* and the biomedical summaries, we achieved better estimations on both the two measures. For the *Readability* estimation, the Sum-QE that trained using multi-task learning was the best achieving 0.198 mae from the gold scores. For the *Repetition* estimation the best model was the one trained using single-task learning (SUM-QE REP (ST)) and achieved 0.202 mae from the manual scores. Hence, we conclude that the Sum-QE model can also be used to assess the quality of biomedical summaries when we train it at the scores that we want to estimate.

5 Content estimation for text summarization

5.1 Introduction

In this chapter, we present an extension to the work of Xenouleas et al. (2019) which also accounts for content preservation in the automatically generated summaries. The difference from the quality estimation that we described previously in Section 4 is in that case we did not need any relevant document to compare with, since in order to observe whether a summary has good quality characteristics, we could rely only on the system summary. However, in order to evaluate a system summary for content retention, we need a reference summary, either a human-written summary or the source document, to be compared with. We analyze the experiments conducted in order to check whether the key points of the source document(s) are captured in the system generated summary. Specifically, we try to predict the content-based scores provided along with the datasets presented in Section 3. These scores are the *Information Recall* and *Information Precision* of the BioASQ dataset alongside the *Informativeness* and *Relevance* of the NEWSROOM dataset. Actually, we mentioned that these measures are semantically similar to each other. The *Information Recall* which captures the extent to which a summary contains all the necessary information, corresponds to the *Informativeness* measure how well the summary captures the key points of the article. Additionally, the *Information Precision* measurement which captures the extent to which the summary does not contain irrelevant information, corresponds to the *Relevance* which captures whether the details provided by the summary are consistent with details in the article or not.

5.2 Methods

In contrast to quality estimations (chapter 4), we did not divide the methods we experimented with into categories to distinguish the news from the biomedical summaries content estimation. All the methods were evaluated in both domains.

ROUGE: We started by calculating the ROUGE scores (Lin and Hovy, 2003; Lin, 2004) for each summary regardless of the domain (news or biomedical) it belongs. Although ROUGE focuses on surface similarities between peer and human-written (reference) summaries, we would expect measurements like *Informativeness* and *Information Recall* to be captured, to some extent, by ROUGE versions based on long n-grams or longest common subsequences. The versions appear in Table 5.1 (ROUGE-W-STOP-P and ROUGE-W-STEM-STOP-P) are the ones that perform best on each measurement of the NEWSROOM dataset. In the Table 5.2 we can see the versions (ROUGE-1-STEM-R and ROUGE-1-STEM-STOP-R) that perform best on the validation data, evaluated though on the test data of the BioASQ dataset. The best versions are obtained among those considered by Graham (2015).

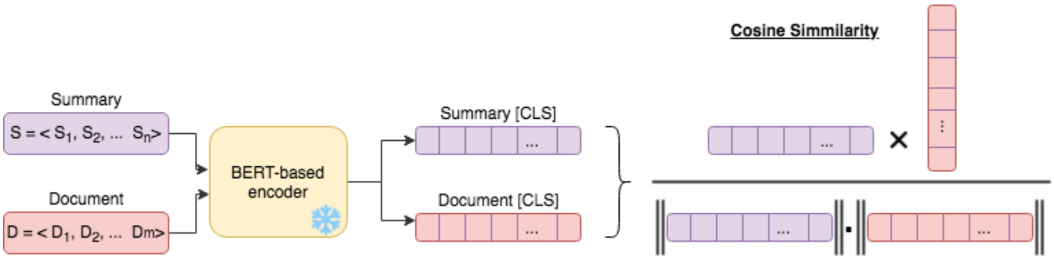

Figure 5.1.: Illustration of cosine similarity between the summary's and document's embedding using SENTENCE-BERT transformers to encode the summary and the document.

SBERT Cosine Similarity (CS): It is very common to read a summary that does not include as many words from the article and has replaced them with synonyms or paraphrases. In these cases, the n-gram based measures like ROUGE (Lin and Hovy, 2003; Lin, 2004) and BLEU (Papineni et al., 2002) cannot perform good content estimations since the common n-grams are limited. For this reason, we relied on the word embeddings (Mikolov et al., 2013; Pennington et al., 2014), which are learned token representations and can encapsulate many different properties of the words. Specifically, we relied on the contextual embeddings from the BERT model (Devlin et al., 2019) which can generate different vector representations for the same word in different sentences depending on the surrounding words, which form the context of the target word. By using them, we will be able to match paraphrases (instead of exact matching), to capture distant dependencies and also to penalize semantically-critical ordering changes. We used the SENTENCE-BERT (Reimers and Gurevych, 2019) transformers (9 in total) to encode our summaries and articles in order to compare them and measure the shared content.[1] Starting from the simplest approach, we calculated the cosine similarity between the source document's and summary's embeddings produced by a sentence-transformer without any further training. On the BioASQ dataset, we had more than one source documents so we calculated the similarity of the summary's embedding with each source document and assigned the average similarity as an estimation to the content-measure. In all cases, the whole text was fed into the encoder to produce a single vector. We can see in Figure 5.1 an illustration of the whole procedure using only one source document but the procedure can be generalized easily by averaging the produced scores for each available source document. The whole summary $S =< S_1, S_2...S_n >$ and document $D =< D_1, D_2...D_m >$, with S_i and D_i

[1] We experimented with the following transformers: BERT-BASE-NLI-MEAN-TOKENS, BERT-BASE-NLI-MAX-TOKENS, BERT-BASE-NLI-CLS-TOKEN, BERT-LARGE-NLI-MEAN-TOKENS and BERT-LARGE-NLI-CLS-TOKEN

denoting the vocabulary id of each byte pair encoding (BPE), were treated as one big sentence, hence the $[CLS]$ embeddings produced by the model, correspond to the whole summary and document respectively. As known, the BERT model has a restriction on the input length to be not more than 512 BPEs but in our case the system-generated summaries were not larger than this threshold and most of the source documents were not either. It should be mentioned that the cosine similarity produces scores in the range [-1, 1] so in order to compute the MAE correctly, we normalized them to [0, 1]. In Table 5.1 we can see the transformer achieved the best estimations evaluating on the NEWSROOM dataset. In Table 5.2 the performance of the transformer achieved the best estimations on the validation set of BioASQ when was used to estimate the corresponding content-based measurement on the test set can be found. The results from all the transformers used as encoders can be found in the Appendix (Section A.2.1).[2]

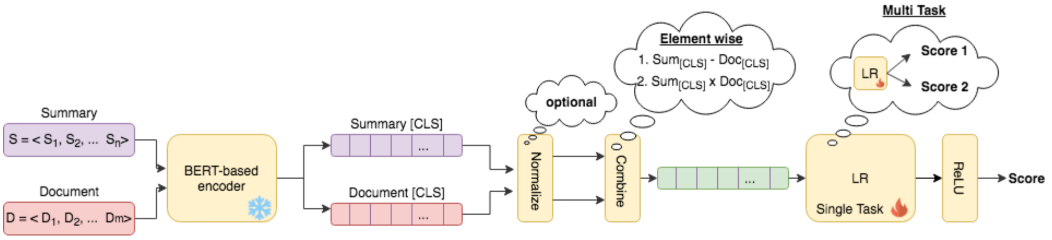

Figure 5.2.: Pre-processing of the summary's and document's embedding before being given as input to the linear regressor.

Sbert LR: We tried a better way to compare the embeddings produced by the SENTENCE-BERT transformers adding supervision. Keeping frozen the same encoders as before, we trained a linear regressor (LR) to learn to weigh the features of the embeddings produced by the transformers according to the measurement that we want to estimate. The whole summary and relevant documents were treated again as one big sentence, hence we had two $[CLS]$ embeddings that we wanted to combine appropriately and produce one vector which would be the input to the LR. Inspired by the cosine-similarity measure, which has the product of the L_2 norms of the embeddings on the denominator, we added a normalization layer, which (optionally) normalizes the two embeddings dividing their features by the L_2 norm of the corresponding vector. Additionally, after the normalization layer, inspired by Reimers and Gurevych (2019) who trained the SBERT using the concatenation of the sentence embeddings u, v alongside the element wise difference $|u - v|$, of sentence A and sentence B respectively, we added a "combine" layer. This layer is responsible to combine the two vectors and produce a singe vector that will form the input of the LR. We experimented with two combination strategies, the element-wise difference $\vec{D}_{[CLS]} - \vec{S}_{[CLS]}$ and the element-wise multiplication $\vec{D}_{[CLS]} * \vec{S}_{[CLS]}$. Both strategies, produce one vector which encapsulates the information of the comparison and this is provided as input to the LR to learn to weigh appropriately its features. Similarly to quality estimation (Section 4.2), we trained the linear regressor using the single-task learning,

[2] In Table A.2 can be found the results of the NEWSROOM dataset and in Tables A.4, A.3 the results of the test and validation sets of the BioASQ dataset respectively.

trying to learn to predict one score each time, and multi-task learning trying to learn to predict both of the content-base measurements of each dataset (*Informativeness* and *Relevance* on the NEWSROOM and *Information Precision* and *Information Recall* on the BIOASQ dataset) simultaneously. Finally, the predicted score(s) of the linear regressor pass through a $ReLU$ layer in order to end up with scores in [0,1]. In Figure 5.2 we can see an illustration of the described pipeline. We trained the regressor using the combinations of the following options:

1. Transformer model
2. Normalization (NORM) or not
3. Combination of vectors using elementwise difference (DIFF) or multiplication (MULT)
4. Training using single-task (ST) or multi-task learning (MT)

Similarly to the previous experiments, in Tables 5.1, 5.2 we can see the performance of the transformers that best performed on the NEWSROOM dataset and the transformer that best performed on the validation set of the BIOASQ dataset, evaluated though on the test set, respectively. The results of the trained regressor using all the possible options presented can be found in the Appendix (Section A.2.2).[3]

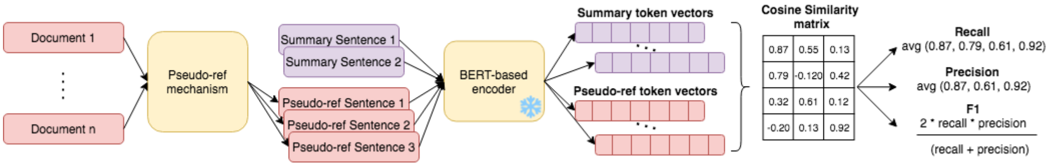

Figure 5.3.: Illustration of SUPERT pipeline.

SUPERT & ALT. SUPERT: Turning back to the unsupervised evaluation measures, we tried the SUPERT model for our content evaluation. Gao et al. (2020) developed an unsupervised, multi-document and reference-free evaluation measure which can be used either on its own to evaluate system summaries or as a reward function to guide a neural, reinforcement learning based summarizer, to generate summaries. Due to the lack of human written summaries, they developed also a mechanism which constructs a pseudo-reference summary using the source documents in order to be compared with a system summary. Gao et al. (2020) tried many different methods to select the salient sentences and compose a pseudo-reference summary. They started by using simple heuristics, concatenating the first n sentences of each source document treating this as the pseudo-reference. They used also some graph based heuristics (Erkan and Radev, 2004; Zheng and Lapata, 2019) where each vertex represents a sentence from the source document(s) and the weight of each edge corresponds to the similarity of the corresponding sentence pair. The pseudo-reference in these methods was built by extracting the semantically "central" sentences from each

[3] Tables A.5, A.6 correspond to the results of the *Informativeness* and *Relevance* estimations of the NEWSROOM dataset. Tables A.8, A.7 correspond to the results of the *Information Precision* estimation on the test and validation sets respectively of the BIOASQ dataset and Tables A.10, A.9 correspond to the results of the *Information Recall* estimation on the test and validation sets of the same dataset respectively.

clique observed in the graph avoiding including two sentences of the same clique which might cause including in the summary sentences reporting very similar information. Finally, they used the cosine similarity of SENTENCE-BERT embeddings in order to obtain the least semantically similar sentence pairs and concatenate them to compose a summary. Unfortunately, none of the above methods had better results than the simple heuristics and they ended up to propose a mechanism that concatenates the first 10 sentences of the original documents treating this as the pseudo reference and comparing it with the candidate system summary. Also, for news articles, using the first sentences of the article as a reference may be reasonable, because news articles usually start by providing a summary. In Figure 5.3 we can see an illustration of the SUPERT pipeline. After the selection of the salient sentences from the source documents, each sentence of the pseudo-reference and the candidate summary, pass one by one through a shared SENTENCE-BERT transformer which produces a vector for each token of the sentence. Aggregating all the token vectors from the pseudo-reference and the candidate summary, it calculates the cosine-similarity matrix with shape (#reference tokens, #summary tokens). This matrix shows how similar each word of the reference is to each word of the candidate summary and vice-versa. In the end, using the cosine similarity matrix, three measures can be calculated:

- **PRECISION**: matches each token of the candidate summary to a token in the reference by observing the most similar pair and averaging the best similarities over the summary tokens.
- **RECALL**: matches each token of the reference to a token in the candidate summary, by observing the most similar pair, and averaging the best similarities over the tokens of the reference.
- **F1**: the harmonic mean of the precision and recall.

We experimented with all of the three measures that Gao et al. (2020) provided and we also changed a little bit the published code to conduct some more experiments. Firstly, apart from the original SENTENCE-BERT transformer that SUPERT uses, we tried all the transformers that were used in the above paragraph where the experiments with SENTENCE-BERT are analyzed.[4] Secondly, we conducted some experiments using the pseudo-reference mechanism described but we also turned it off to inspect whether giving the concatenation of all the documents or the real reference as pseudo-reference, would achieve better results. Also, on the BioASQ dataset, we had some snippets (relevant word sequences from the source documents annotated by the experts) and we also checked whether the concatenation of them, treated as a pseudo-reference, can achieve better results. In the appendix (Section A.2.3), the results are shown, from all the different options we followed:

[4] SUPERT uses BERT-LARGE-NLI-STSB-MEAN-TOKENS as encoder.

1. Transformer model.
2. Document type (SD: concatenation of all the source documents, REF: the real reference, MECH: pseudo-reference produced by the published mechanism).
3. Measures (PREC: precision, REC: recall and F2)

In Tables 5.1, 5.2 we can see the original version of SUPERT along with the alternative versions that best performed on the NEWSROOM dataset and on the validation set of the BioASQ dataset, evaluated though on the test set, respectively. In the Appendix (Section A.2.3), the results from all the possible combination of the above-mentioned options, can be found.[5]

5.3 Experimental Results

We used the same methods we used to evaluate a quality-base measure and these are:

1. The micro-average of MAE (Equation 2.4) comparing all the predictions to the human scores of the summaries, regardless of the document (or the question) they summarise (or answer). This is also our main evaluation factor and the comparison between each measurement was conducted by comparing the MAE scores achieved from the human scores.
2. The macro-average of MAE (Equation 2.4) across documents (or questions).[6] Alongside the macro-average of MAE, we present the standard error of the mean (SEM) after the ± symbol (Equation 2.5).
3. The Spearman's ρ (Equation 2.2), Kendall's τ (Equation 2.1) and Pearson's r (Equation 2.3) correlations at the Document level (Section 2.3). Alongside with the correlations, we also present the standard error of the mean (SEM) after the ± symbol (Equation 2.5).

Unlike the quality estimation (Section 4.3), all the methods we experimented, produce output in the range of [0, 1], so no further processing was required in order to have accurate evaluation. We should mention that the results that we see in Table 5.2 are the performance of the corresponding measures/models that best performed on the validation data.

Starting from ROUGE, as we expected, the versions based on the Longest Common Subsequences (LCS) statistics, ROUGE-L and ROUGE-W, are the best, among those considered

[5] In Tables A.11, A.12 the results from the *Informativeness* and *Relevance* estimation of the NEWSROOM dataset can be found, respectively. In Tables A.15, A.16 the results from the *Information Recall* estimation on the validation and test sets of the BioASQ dataset are shown, respectively. In Tables A.13, A.14 the results from the *Information Precision* estimation on the validation and test sets of the BioASQ dataset are shown, respectively.

[6] On the NEWSROOM dataset, all the 7 systems constructed a summary for all the 60 articles, so none is missing. Therefore, the MAE on the micro is the same with MAE on the macro level.

by Graham (2015), for estimating the *Relevance* and *Informativeness* scores of the NEWSROOM dataset respectively. However, this does not hold for the *Information Precision* and *Information Recall* estimations of the BioASQ dataset. ROUGE -1 achieved the best results and this probably happens because, based only on unigrams overlaps, there may be individual biomedical terms on which we can be relied on to make sure that the summary contains all the necessary information (*Information Recall*) and does not contain irrelevant information (*Information Precision*). Moving on, to the experiments we conducted using the SENTENCE-BERT transformers, we can observe that calculating the cosine similarity (CS) between the embeddings of the document and the summary, did not improve the estimations of *Informativeness* and *Relevance* of the news-domain summaries compared to ROUGE.[7] On the contrary, on the biomedical domain summaries, the estimations of the *Information Precision* and *Information Recall* are improved compared to ROUGE.[8] This means that there may not be as many shared n-grams between the biomedical summaries and documents in order to let ROUGE work better. However, the different terms the two texts have included, are semantically close each other, which can be captured by the SENTENCE-BERT embeddings, hence the cosine similarity between them can provide a better estimation. In the experiments where we trained a linear regressor using as input the combination of the SENTENCE-BERT embeddings from the summary and document, trying to learn to predict the corresponding measurement(s), it seems that the models cannot perform accurate estimations for none of the above mentioned measures except for the *Information Recall* of the BioASQ dataset. [9] [10] In fact, at the *Information Recall* estimation, the linear regressor achieved a small MAE from the manual scores, equal to 0.161 but was not the best. Finally, the original SUPERT model was not so good at its estimations but using the "alternative" versions (ALT. SUPERT) described above, we managed to achieve the lowest MAE from the manual scores. [11] [12]

[7] The transformers that best performed on the estimations of *Informativeness* and *Relevance* can be found in Table A.2 as BERT-BASE-NLI-CLS-TOKEN (CS) and BERT-BASE-NLI-MEAN-TOKENS (CS), respectively.

[8] The transformers that best performed on the estimations of the *Information Precision* and *Information Recall* can be found in Table A.4 as BERT-BASE-NLI-MEAN-TOKENS (CS) and BERT-BASE-NLI-MAX-TOKENS (CS), respectively.

[9] The transformers that best performed on the estimations of the *Informativeness* and *Relevance* can be found as BERT-BASE-NLI-MEAN-TOKENS (MT-MULT) in Table A.5 and as BERT-BASE-NLI-MEAN-TOKENS (MT-MULT) in Table A.6.

[10] The transformers that best performed on the estimations of the *Information Precision* and *Information Recall* can be found as BERT-BASE-NLI-MEAN-TOKENS (MT-DIFF) in Table A.8 and as BERT-LARGE-NLI-CLS-TOKEN (MT-MULT) in Table A.10.

[11] The alternative version of SUPERT that best performed on the estimations of the *Informativeness* and *Relevance* can be found as BERT-BASE-NLI-MAX-TOKENS (MECH-REC) in Table A.11 and as BERT-BASE-NLI-MEAN-TOKENS (SD-F1) in Table A.12, respectively.

[12] The alternative version of SUPERT that best performed on the estimations of the *Information Precision* and *Information Recall* can be found as BERT-LARGE-NLI-STSB-MEAN-TOKENS (SD-PREC) in Table A.14 and as BERT-BASE-NLI-MAX-TOKENS (SNIP-F1) in Table A.16, respectively.

		NEWSROOM content estimations			
	Method	MAE	ρ	τ	r
Informativeness	ROUGE-W-STOP-P	0.217	0.434 ± 0.044	0.337 ± 0.300	0.272 ± 0.289
	SBERT CS	0.279	**0.735 ± 0.024**	**0.627 ± 0.026**	0.767 ± 0.023
	SBERT LR	0.185	0.439 ± 0.043	0.351 ± 0.037	0.536 ± 0.046
	SUPERT (RECALL)	0.194	0.715 ± 0.027	0.611 ± 0.029	0.779 ± 0.022
	ALT. SUPERT	**0.118**	0.711 ± 0.029	0.611 ± 0.030	**0.777 ± 0.023**
Relevance	ROUGE-L-STEM-STOP-P	0.222	0.440 ± 0.045	0.366 ± 0.039	0.551 ± 0.041
	SBERT CS	0.204	0.625 ± 0.030	0.522 ± 0.029	0.780 ± 0.021
	SBERT LR	0.179	0.382 ± 0.047	0.289 ± 0.040	0.556 ± 0.044
	SUPERT (PRECISION)	0.137	0.530 ± 0.039	0.451 ± 0.036	0.726 ± 0.030
	ALT. SUPERT	**0.106**	**0.636 ± 0.027**	**0.526 ± 0.027**	**0.794 ± 0.022**

Table 5.1.: Mean Absolute Error (MAE) alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (SEM) between human (*Informativeness* and *Relevance*– Section 3.2) and automatic measures (ROUGE, SBERT CS, SBERT LR, SUPERT and ALT. SUPERT version) at the document level. The scores of the SBERT CS, SBERT LR, ALT. SUPERT correspond to the best versions from the ones that we experimented with. The results for each version can be found in Appendix A.

		BIOASQ (2019 Test) content estimations				
		Micro	*Macro*			
	Method	MAE	MAE	ρ	τ	r
Information Precision	ROUGE -1-STEM-R	0.315	0.328 ± 0.008	0.082 ± 0.045	0.075 ± 0.043	0.131 ± 0.044
	SBERT CS	0.208	0.201 ± 0.005	-0.045 ± 0.035	-0.048 ± 0.033	0.106 ± 0.038
	SBERT LR	0.268	0.262 ± 0.006	0.197 ± 0.032	0.186 ± 0.030	0.255 ± 0.034
	SUPERT (PRECISION)	**0.196**	**0.188 ± 0.006**	**0.216 ± 0.032**	**0.206 ± 0.031**	**0.281 ± 0.033**
	ALT. SUPERT	**0.196**	0.189 ± 0.005	0.184 ± 0.033	0.170 ± 0.031	0.270 ± 0.033
Information Recall	ROUGE -1-STEM-STOP-R	0.256	0.266 ± 0.009	**0.726 ± 0.023**	**0.695 ± 0.023**	**0.738 ± 0.022**
	SBERT CS	0.179	0.178 ± 0.007	0.611 ± 0.021	0.566 ± 0.020	0.689 ± 0.020
	SBERT LR	0.161	0.163 ± 0.008	0.649 ± 0.023	0.635 ± 0.023	0.660 ± 0.023
	SUPERT (RECALL)	0.337	0.330 ± 0.006	0.612 ± 0.021	0.569 ± 0.020	0.686 ± 0.021
	ALT. SUPERT	**0.160**	**0.162 ± 0.006**	0.640 ± 0.019	0.594 ± 0.018	0.721 ± 0.020

Table 5.2.: Mean Absolute Error (MAE) on the *micro* level and *macro* MAE alongside the Spearman's ρ, Kendall's τ, Pearson's r correlations ± Standard Error of the Mean (SEM) between human (*Information Precision* and *Information Recall*– Section 3.3) and automatic measures (ROUGE, SBERT CS, SBERT LR, SUPERT and ALT. SUPERT version) at the document level. The scores of the SBERT CS, SBERT LR, ALT. SUPERT correspond to the versions that best performed in the validation data, evaluated though in the test set of BIOASQ dataset. The results for each version that we experimented with can be found in Appendix A.

Conclusions and Future Work 6

6.1 Conclusion

In this thesis, we addressed the task of automatic evaluation of system generated summaries considering two dimensions: the quality of the system generated summary which focuses on quality aspects that reflect the coherency, the repetition, the fluency and the readability of the generated text and the extent to which the generated summaries encapsulate the most important information expressed in the source document(s). We focused on two datasets, NEWSROOM and BioASQ, which contain summaries alongside human evaluations from the news and biomedical domain respectively. We compared the assigned scores of each automatic measure that we evaluated, to the manual scores by computing either the correlation they have or the mean absolute error between them. The mean absolute error was our main focus since the ultimate goal was to propose a measure which can be used independently to evaluate a single-summary, producing accurate estimations close to the manual scores without relying on the ranking of the evaluated summaries. However, we conducted an analysis including the correlations at the document level which indicates that the ranking order of the systems per document, extracted by the automatic scores, is similar to the one extracted by the manual scores.

At the quality estimation of the news domain, we used the automatically produced scores provided by Grusky et al. (2018) (*Coverage* and *Density*) and the predictors of Sum-QE (Xenouleas et al., 2019) trained on DUC data. We checked how each one performs separately and then we combined them using an average or by training a linear regressor to learn to weight them appropriately. We concluded that the linear combination of all the above-mentioned scores achieved the best estimations in terms of absolute error. Similarly, on the quality estimation of the biomedical summaries, we also used our previous Sum-QE model (Xenouleas et al., 2019) as a starting point. First, we checked whether the predictors of the model, trained on the DUC data, can help estimate the *Readability* and the *Repetition*. We used the predictors of *Grammaticality* ($\mathcal{Q}1$) and the *Non redundancy* ($\mathcal{Q}2$) which are semantically closer to *Readability* and *Repetition*. However the pre-trained predictors could not achieve good estimations so we trained the model from scratch using the biomedical summaries and as ground truth scores, the scores that we want to estimate (*Readability* and the *Repetition*). Therefore, we trained the model using single-task and multi-task learning and we managed to produce better estimations achieving lower absolute error to the manual scores than the pre-trained predictors on the new data. Consequently, the

Sum-QE model can also handle the data from the biomedical domain and can greatly improve the estimations of *Readability* and *Repetition.*

In the contrast to quality estimation, at the content estimation we couldn't use the Sum-QE as a base model since it was developed and trained to capture quality aspects of the generated-texts. We started by calculating the rouge scores of each summary among the versions considered by Graham (2015). We concluded that: (i) there was not a common rouge version that was better to the estimation of all the human-measures; (ii) and the absolute error of the best versions each time was too high from the manual scores. Moving forward, we used the sentence transformers (Reimers and Gurevych, 2019) in several variations using supervision or not. The unsupervised approaches ware to calculate the cosine similarity between the summary and the source document(s) embeddings produced by the frozen sentence transformers and to calculate the recall, the precision and the f1 scores as defined by Gao et al. (2020). In order to calculate these scores, we had to construct a cosine similarity matrix between the word embeddings of the summary and the compared document, produced by the supert model using as reference article: (a) the pseudo-reference, produced by the published mechanism of Gao et al. (2020); (b) the source document(s); or (c) the human-written summary. For the supervised approach, we trained a linear regressor using as input a vector produced by the comparison (element-wise subtraction or multiplication) of the sentence-bert embeddings that correspond to the summary and the source document. We ended up that the unsupervised approach, using the supert pipeline with slight changes explained in Section 5.2, was the best at the content estimation of the news or the biomedical data.

6.2 Future Work

In future work, we plan to experiment more with other models even by adjusting them to a current experimental setup or training them from scratch. SciBERT (Beltagy et al., 2019) and BioBERT (Lee et al., 2019) are some of these which are domain-specific language representation models based on bert (Devlin et al., 2019), pre-trained on a large-scale of scientific and biomedical data respectively. Adjusting the above mentioned models to our current experimental setup, we can get better representations, especially for the biomedical data, even keeping them frozen, which can encapsulate more information from the summary and source document(s) and the comparison between them may improve the content estimation. Also, these models can be fine tuned to our biomedical data trying to predict the manual scores (e.g. *Information Recall*) producing even better embeddings which can be utilized appropriately to other tasks. A different approach which would also be interesting to examine is to use the Sum-QE model trained on DUC data, having learned to predict a specific measure (e.g., *Grammaticality*) and fine tune it on the NEWSROOM data, trying to predict a similar measure (e.g., *Fluency*) in order to observe whether it can improve even more the estimations utilizing the pre-trained weights from the news data.

Bibliography

Banerjee, Satanjeev and Alon Lavie (June 2005). “METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments”. In: *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization.* Ann Arbor, Michigan: Association for Computational Linguistics, pp. 65–72 (cit. on p. 15).

Beltagy, Iz, Kyle Lo, and Arman Cohan (Nov. 2019). “SciBERT: A Pretrained Language Model for Scientific Text”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP).* Hong Kong, China: Association for Computational Linguistics, pp. 3615–3620 (cit. on p. 31).

Bojar, Ondrej, Rajen Chatterjee, Christian Federmann, et al. (Aug. 2016). “Findings of the 2016 Conference on Machine Translation”. In: *Proceedings of the First Conference on Machine Translation.* Berlin, Germany: Association for Computational Linguistics, pp. 131–198 (cit. on p. 15).

Bojar, Ondrej, Rajen Chatterjee, Christian Federmann, et al. (Sept. 2017). “Findings of the 2017 Conference on Machine Translation (WMT17)”. In: *Proceedings of the Second Conference on Machine Translation.* Copenhagen, Denmark: Association for Computational Linguistics, pp. 169–214 (cit. on p. 15).

Bowman, Samuel R., Gabor Angeli, Christopher Potts, and Christopher D. Manning (Sept. 2015). “A large annotated corpus for learning natural language inference”. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing.* Lisbon, Portugal: Association for Computational Linguistics, pp. 632–642 (cit. on p. 44).

Callison-Burch, Chris, Miles Osborne, and Philipp Koehn (Apr. 2006). “Re-evaluating the Role of Bleu in Machine Translation Research”. In: *11th Conference of the European Chapter of the Association for Computational Linguistics.* Trento, Italy: Association for Computational Linguistics (cit. on pp. 1, 8).

Chaganty, Arun, Stephen Mussmann, and Percy Liang (July 2018). “The price of debiasing automatic metrics in natural language evalaution”. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* Melbourne, Australia: Association for Computational Linguistics, pp. 643–653 (cit. on p. 10).

Dang, Hoa Trang (2006a). “DUC 2005: Evaluation of Question-focused Summarization Systems”. In: *Proceedings of the Workshop on Task-Focused Summarization and Question Answering*. SumQA ’06. Sydney, Australia, pp. 48–55 (cit. on p. 11).

Dang, Hoa Trang (2006b). “Overview of DUC 2006”. In: *Proceedings of the Document Understanding Workshop at HLT-NAACL 2006*. Vol. 2006. Brooklyn, NY, USA (cit. on p. 11).

Denkowski, Michael and Alon Lavie (June 2014). “Meteor Universal: Language Specific Translation Evaluation for Any Target Language”. In: *Proceedings of the Ninth Workshop on Statistical Machine Translation*. Baltimore, Maryland, USA: Association for Computational Linguistics, pp. 376–380 (cit. on pp. 9, 15).

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (June 2019). “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186 (cit. on pp. 9, 23, 31).

Dorr, Bonnie, David Zajic, and Richard Schwartz (2003). “Hedge Trimmer: A Parse-and-Trim Approach to Headline Generation”. In: *Proceedings of the HLT-NAACL 03 on Text Summarization Workshop - Volume 5*. HLT-NAACL-DUC ’03. USA: Association for Computational Linguistics, pp. 1–8 (cit. on p. 3).

Erkan, Günes and Dragomir R Radev (2004). “Lexrank: Graph-based lexical centrality as salience in text summarization”. In: *Journal of Artificial Intelligence Research* 22, pp. 457–479 (cit. on pp. 3, 25).

Gao, Yang, Wei Zhao, and Steffen Eger (July 2020). “SUPERT: Towards New Frontiers in Unsupervised Evaluation Metrics for Multi-Document Summarization”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 1347–1354 (cit. on pp. 9, 25, 26, 31, 53).

Graham, Yvette (Sept. 2015). “Re-evaluating Automatic Summarization with BLEU and 192 Shades of ROUGE”. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, pp. 128–137 (cit. on pp. 8, 23, 28, 31).

Grusky, Max, Mor Naaman, and Yoav Artzi (June 2018). “Newsroom: A Dataset of 1.3 Million Summaries with Diverse Extractive Strategies”. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 708–719 (cit. on pp. 12, 13, 15, 16, 30).

Guo, Yinuo and Junfeng Hu (Aug. 2019). “Meteor++ 2.0: Adopt Syntactic Level Paraphrase Knowledge into Machine Translation Evaluation”. In: *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*. Florence, Italy: Association for Computational Linguistics, pp. 501–506 (cit. on p. 9).

Hardy, Hardy, Shashi Narayan, and Andreas Vlachos (July 2019). “HighRES: Highlight-based Reference-less Evaluation of Summarization”. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 3381–3392 (cit. on p. 9).

Hermann, Karl Moritz, Tomas Kocisky, Edward Grefenstette, et al. (2015). “Teaching machines to read and comprehend”. In: *Advances in neural information processing systems*, pp. 1693–1701 (cit. on p. 12).

Hsu, Wan-Ting, Chieh-Kai Lin, Ming-Ying Lee, et al. (July 2018). “A Unified Model for Extractive and Abstractive Summarization using Inconsistency Loss”. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, pp. 132–141 (cit. on p. 3).

“Pearson’s Correlation Coefficient” (2008). In: *Encyclopedia of Public Health*. Ed. by Wilhelm Kirch. Dordrecht: Springer Netherlands, pp. 1090–1091 (cit. on p. 5).

Lee, Jinhyuk, Wonjin Yoon, Sungdong Kim, et al. (Sept. 2019). “BioBERT: a pre-trained biomedical language representation model for biomedical text mining”. In: *Bioinformatics* 36.4, pp. 1234–1240 (cit. on p. 31).

Lin, Chin-Yew (July 2004). “ROUGE: A Package for Automatic Evaluation of Summaries”. In: *Workshop on Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, pp. 74–81 (cit. on pp. 1, 8, 22, 23).

Lin, Chin-Yew and Eduard Hovy (2003). “Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics”. In: *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*. Edmonton, Canada: Association for Computational Linguistics, pp. 150–157 (cit. on pp. 1, 8, 22, 23).

Litvak, Marina and Mark Last (2008). “Graph-based keyword extraction for single-document summarization”. In: *Coling 2008: Proceedings of the workshop Multi-source Multilingual Information Extraction and Summarization*, pp. 17–24 (cit. on p. 3).

Lo, Chi-kiu (Sept. 2017). “MEANT 2.0: Accurate semantic MT evaluation for any output language”. In: *Proceedings of the Second Conference on Machine Translation*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 589–597 (cit. on p. 9).

Mihalcea, Rada and Paul Tarau (July 2004). “TextRank: Bringing Order into Text”. In: *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*. Barcelona, Spain: Association for Computational Linguistics, pp. 404–411 (cit. on p. 12).

Mikolov, Tomas, Kai Chen, G. S. Corrado, and J. Dean (2013). “Efficient Estimation of Word Representations in Vector Space”. In: *CoRR* abs/1301.3781 (cit. on p. 23).

Nallapati, Ramesh, Feifei Zhai, and Bowen Zhou (2017). “SummaRuNNer: A Recurrent Neural Network Based Sequence Model for Extractive Summarization of Documents”. In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*. AAAI’17. San Francisco, California, USA: AAAI Press, pp. 3075–3081 (cit. on pp. 3, 12).

Nenkova, Ani and Rebecca Passonneau (May 2004). “Evaluating Content Selection in Summarization: The Pyramid Method”. In: *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*. Boston, Massachusetts, USA: Association for Computational Linguistics, pp. 145–152 (cit. on p. 9).

Nye, Benjamin and Ani Nenkova (May 2015). “Identification and Characterization of Newsworthy Verbs in World News”. In: *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado: Association for Computational Linguistics, pp. 1440–1445 (cit. on p. 3).

Over, Paul, Hoa Dang, and Donna Harman (Nov. 2007). “DUC in Context”. In: *Information Processing & Management* 43.6, pp. 1506–1520 (cit. on p. 11).

Page, Lawrence, Sergey Brin, Rajeev Motwani, and Terry Winograd (Nov. 1999). *The PageRank Citation Ranking: Bringing Order to the Web*. Technical Report 1999-66. Previous number = SIDL-WP-1999-0120. Stanford InfoLab (cit. on p. 12).

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu (July 2002). “BLEU: A Method for Automatic Evaluation of Machine Translation”. In: *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, pp. 311–318 (cit. on pp. 1, 8, 15, 23).

Pennington, Jeffrey, Richard Socher, and Christopher Manning (Oct. 2014). “GloVe: Global Vectors for Word Representation”. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, pp. 1532–1543 (cit. on p. 23).

Puka, Llukan (2011). “Kendall’s Tau”. In: *International Encyclopedia of Statistical Science*. Ed. by Miodrag Lovric. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 713–715 (cit. on p. 5).

Radev, Dragomir R, Hongyan Jing, Małgorzata Styś, and Daniel Tam (2004). “Centroid-based summarization of multiple documents”. In: *Information Processing & Management* 40.6, pp. 919–938 (cit. on p. 3).

Reimers, Nils and Iryna Gurevych (Nov. 2019). “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, pp. 3982–3992 (cit. on pp. 9, 23, 24, 31, 44).

Rush, Alexander M., Sumit Chopra, and Jason Weston (Sept. 2015). “A Neural Attention Model for Abstractive Sentence Summarization”. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, pp. 379–389 (cit. on pp. 1, 12).

Sedgwick, Philip (2014). “Spearman’s rank correlation coefficient”. In: *Bmj* 349, g7327 (cit. on p. 5).

See, Abigail, Peter J. Liu, and Christopher D. Manning (July 2017). “Get To The Point: Summarization with Pointer-Generator Networks”. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, pp. 1073–1083 (cit. on pp. 1, 3, 12).

Sellam, Thibault, Dipanjan Das, and Ankur Parikh (July 2020). “BLEURT: Learning Robust Metrics for Text Generation”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 7881–7892 (cit. on p. 10).

Steinberger, Josef and Karel Jezek (2012). “Evaluation measures for text summarization”. In: *Computing and Informatics* 28.2, pp. 251–275 (cit. on p. 4).

Stent, Amanda, Matthew Marge, and Mohit Singhai (2005). “Evaluating Evaluation Methods for Generation in the Presence of Variation”. In: *Computational Linguistics and Intelligent Text Processing*. Ed. by Alexander Gelbukh. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 341–351 (cit. on pp. 1, 8).

Tsatsaronis, George, Georgios Balikas, Prodromos Malakasiotis, et al. (Apr. 2015). “An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition”. In: *BMC Bioinformatics* 16, p. 138 (cit. on p. 13).

Williams, Adina, Nikita Nangia, and Samuel Bowman (June 2018). “A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference”. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 1112–1122 (cit. on p. 44).

Xenouleas, Stratos, Prodromos Malakasiotis, Marianna Apidianaki, and Ion Androutsopoulos (Nov. 2019). “SUM-QE: a BERT-based Summary Quality Estimation Model”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, pp. 6004–6010 (cit. on pp. 10, 16, 22, 30).

Zhang, Tianyi, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi (2020). “BERTScore: Evaluating Text Generation with BERT”. In: *International Conference on Learning Representations* (cit. on p. 9).

Zheng, Hao and Mirella Lapata (July 2019). “Sentence Centrality Revisited for Unsupervised Summarization”. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 6236–6247 (cit. on p. 25).

Appendix

A

A.1 Additional results for quality estimation

A.1.1 Linear Regression (LR)

Below we can see the training and validation MAE curves of the five-fold cross validation procedure that used to train a linear regressor in order to learn predict the *Fluency* and the *Coherence* scores. The model use as input the predictions of the pre-trained Sum-QE predictors alongside the *Density* and *Coverage* scores. Specifically, we can see 10 figures (5 per measure) with the MAE curves for each one of the six independent training procedures that used to obtain more accurate predictions.

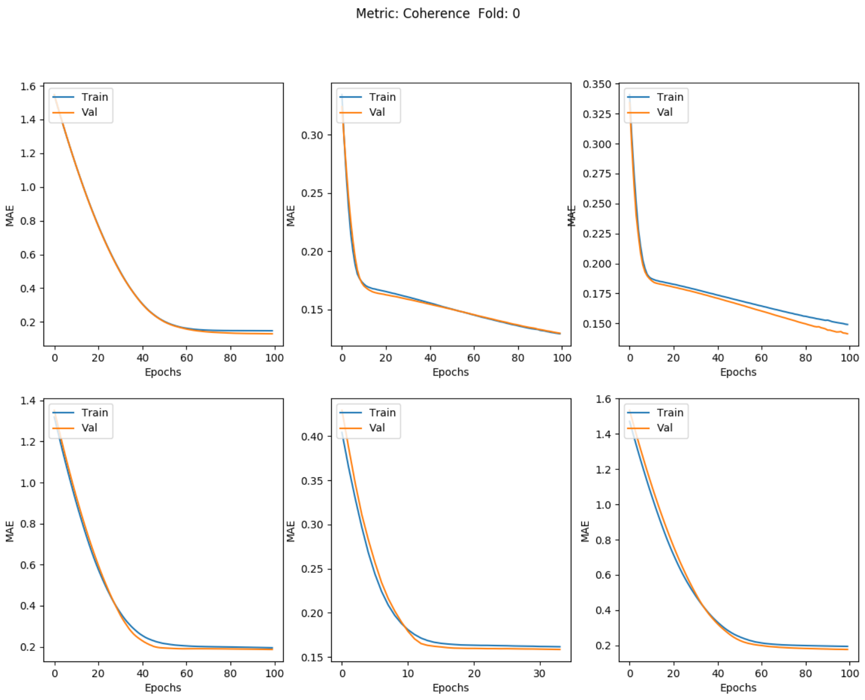

Figure A.1.: The MAE curves from the six independently training procedures of the LR trying to learn to predict the *Coherence* scores of the first fold.

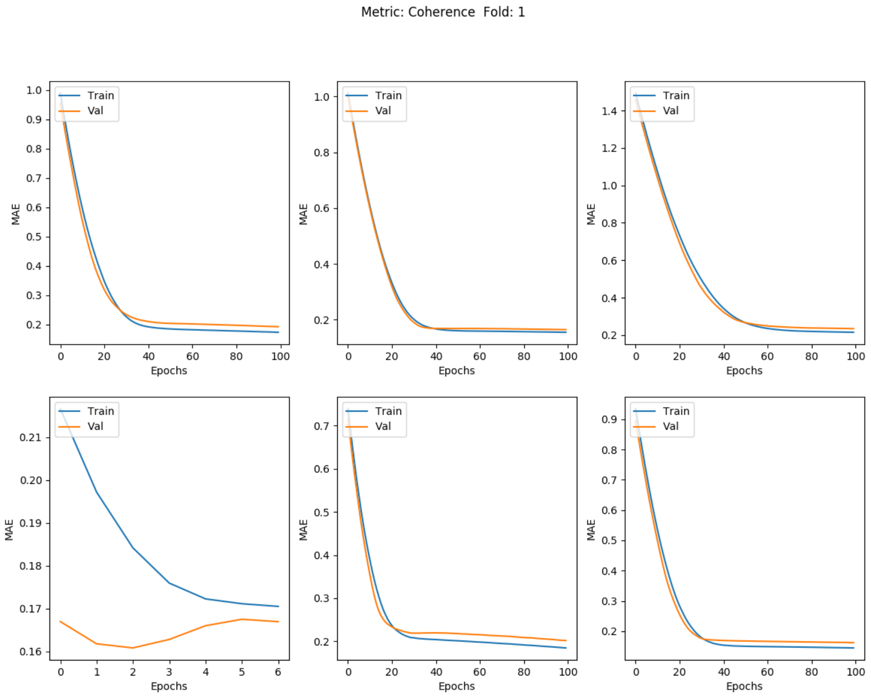

Figure A.2.: The MAE curves from the six independently training procedures of the LR trying to learn to predict the *Coherence* scores of the second fold.

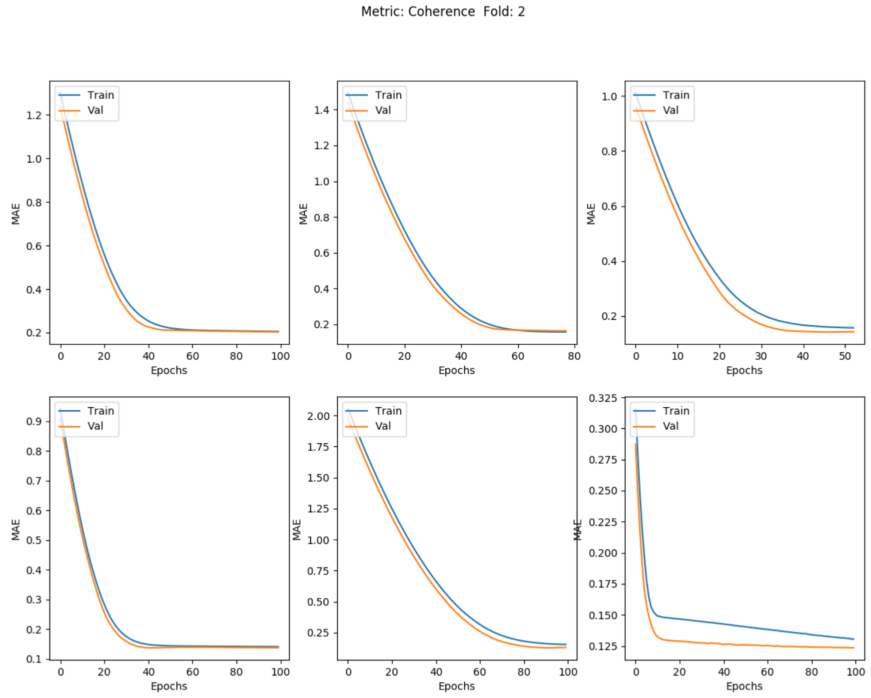

Figure A.3.: The MAE curves from the six independently training procedures of the LR trying to learn to predict the *Coherence* scores of the third fold.

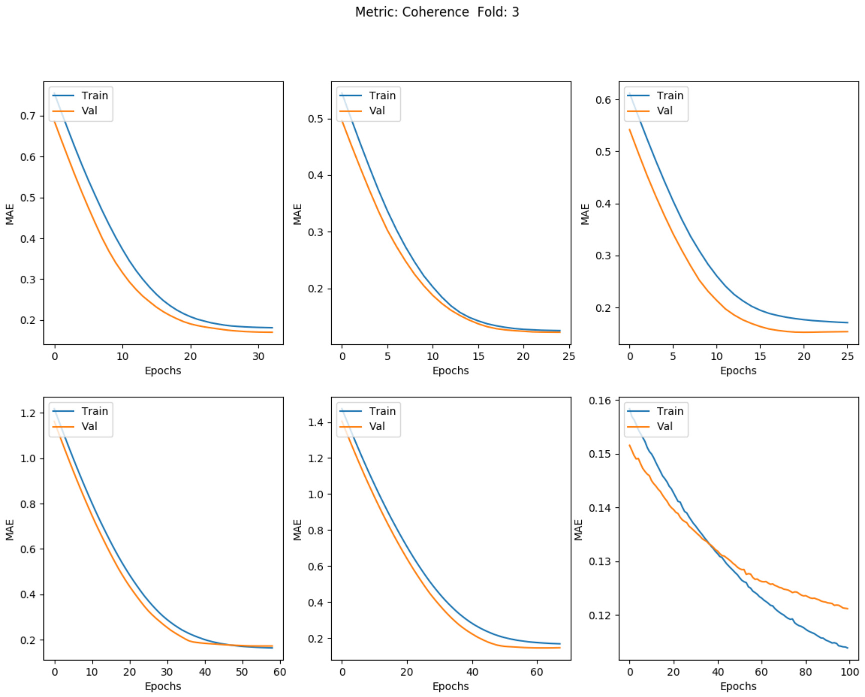

Figure A.4.: The MAE curves from the six independently training procedures of the LR trying to learn to predict the *Coherence* scores of the fourth fold.

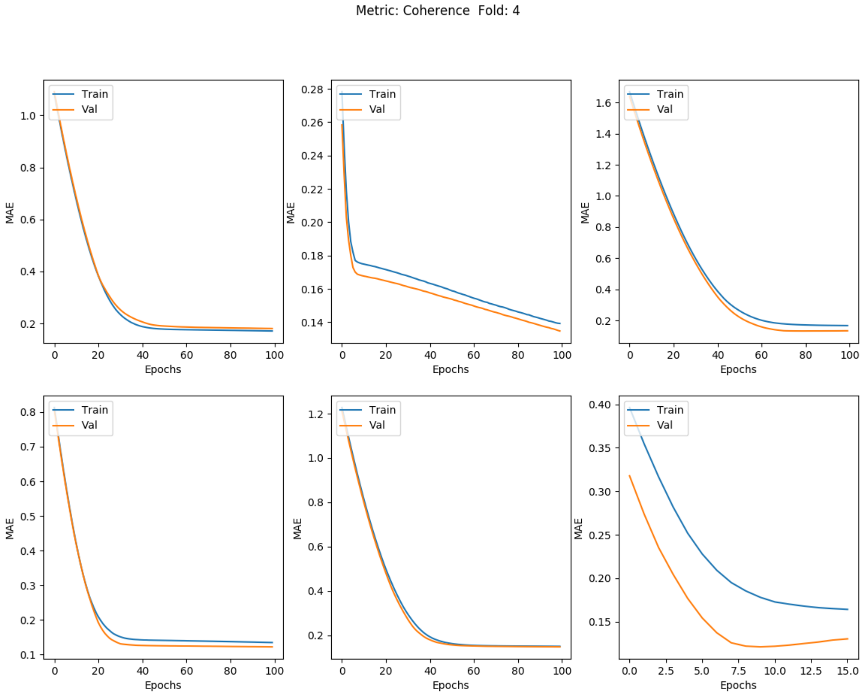

Figure A.5.: The MAE curves from the six independently training procedures of the LR trying to learn to predict the *Coherence* scores of the fifth fold.

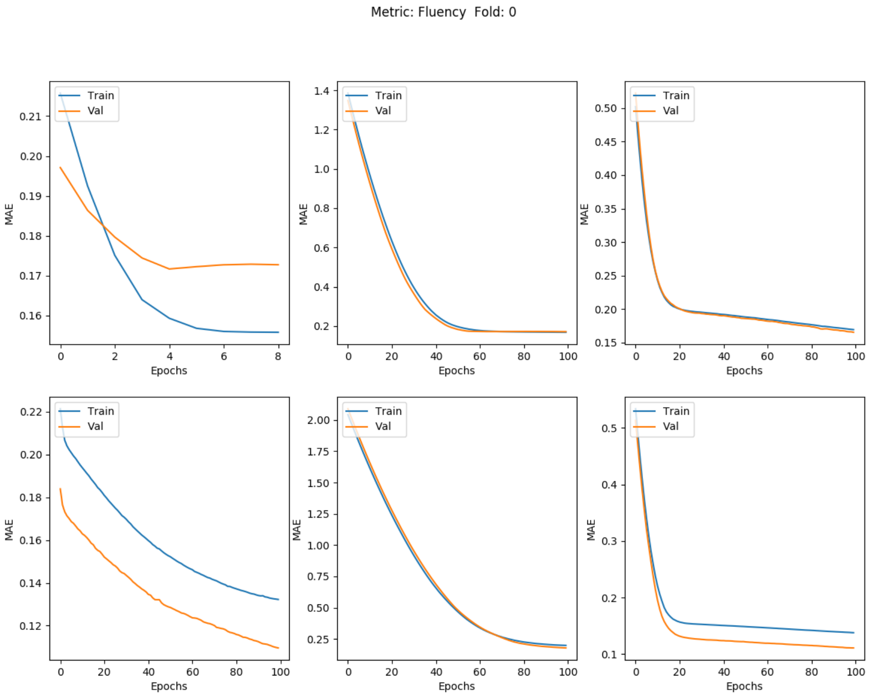

Figure A.6.: The MAE curves from the six independently training procedures of the LR trying to learn to predict the *Fluency* scores of the first fold.

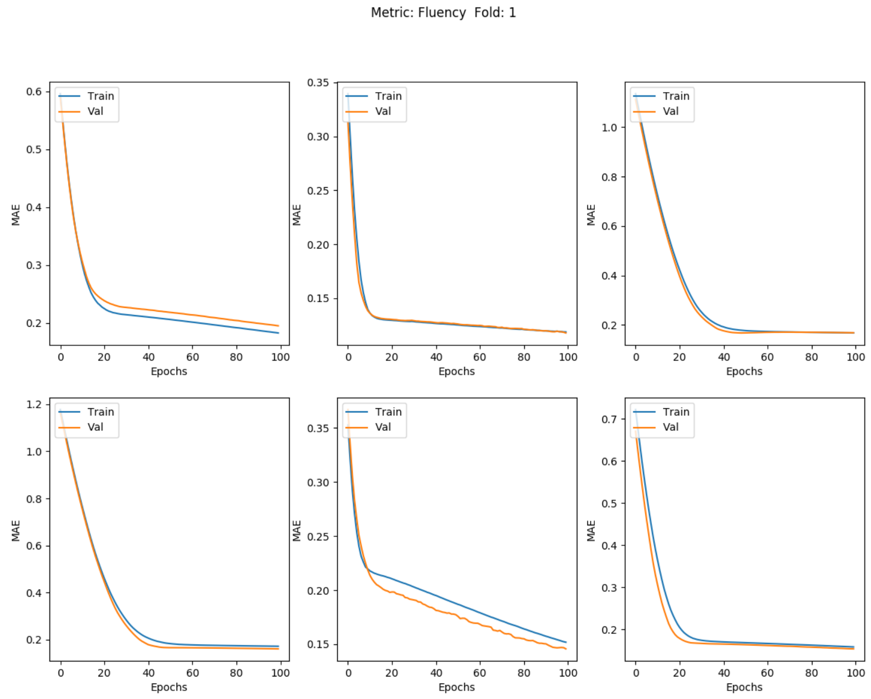

Figure A.7.: The MAE curves from the six independently training procedures of the LR trying to learn to predict the *Fluency* scores of the second fold.

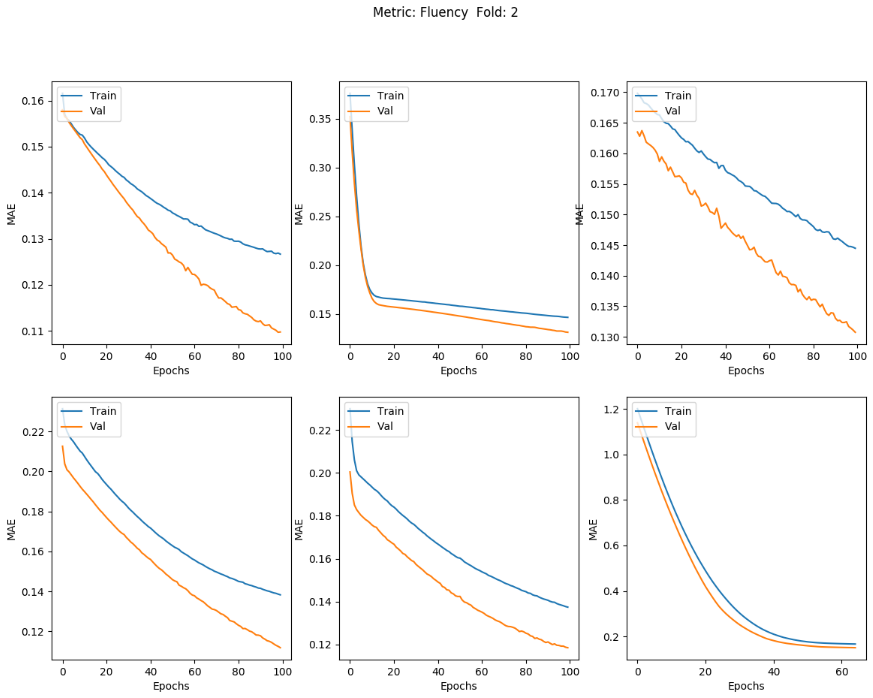

Figure A.8.: The MAE curves from the six independently training procedures of the LR trying to learn to predict the *Fluency* scores of the third fold.

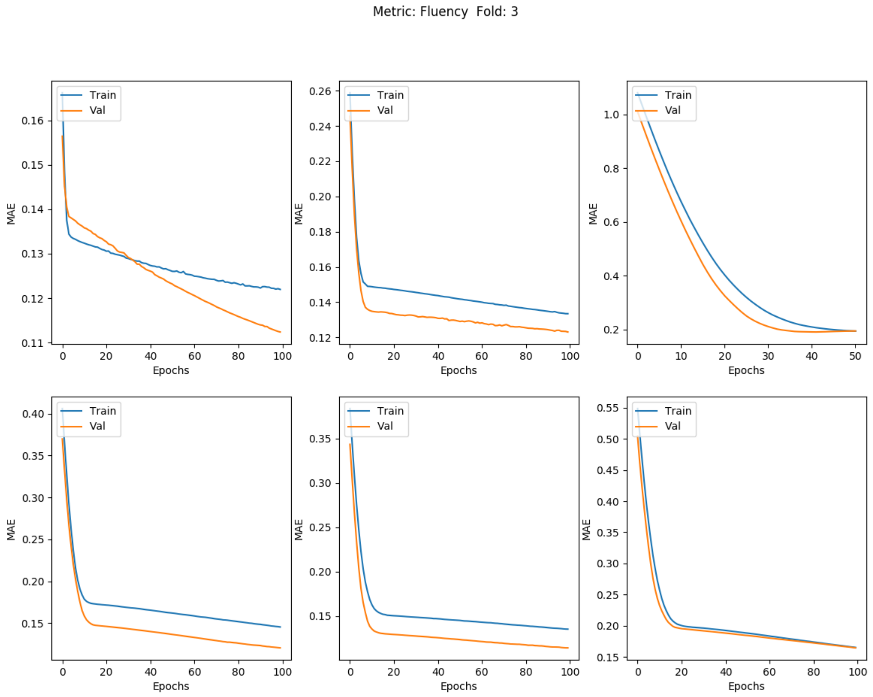

Figure A.9.: The MAE curves from the six independently training procedures of the LR trying to learn to predict the *Fluency* scores of the fourth fold.

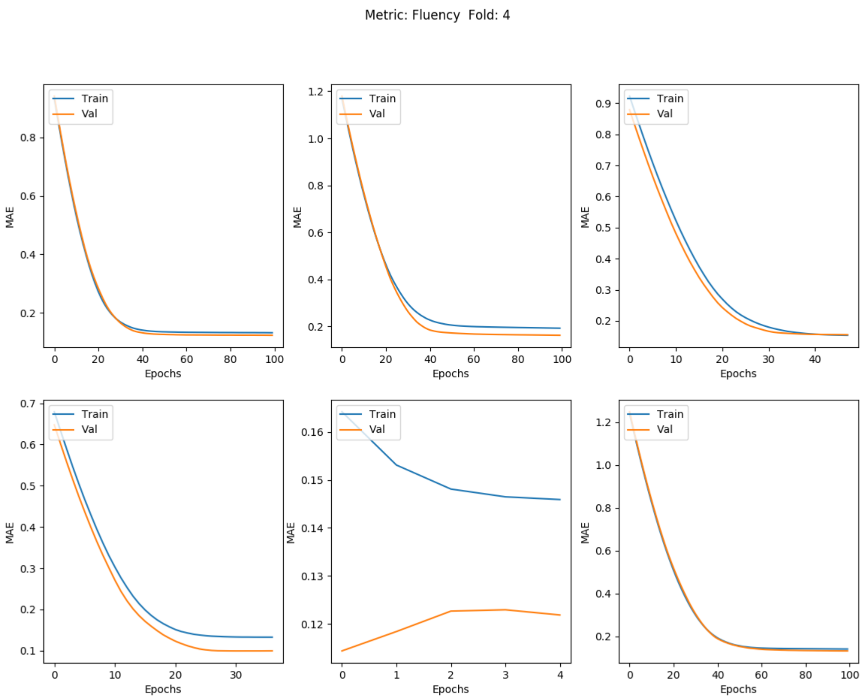

Figure A.10.: The MAE curves from the six independently training procedures of the LR trying to learn to predict the *Fluency* scores of the fifth fold.

A.1.2 Validation results on biomedical quality estimation

Below we can see the results of the measures that we experimented with to assess the quality of a biomedical summary. We can see that the pre-trained predictors of the Sum-QE on the DUC data cannot perform accurate estimations, achieving big MAE from the gold scores. On the other hand the Sum-QE model, trained on the biomedical data trying to predict the scores that we want to estimate, managed to reduce the MAE.

		BioASQ (2018 Val) quality estimations				
		Micro	*Macro*			
	Method	MAE	MAE	ρ	τ	r
READ	SUM-QE $\mathcal{Q}1$ (ST)	0.267	0.274 ± 0.006	0.116 ± 0.021	0.101 ± 0.018	0.171 ± 0.024
	SUM-QE READ (ST)	0.215	0.220 ± 0.005	0.392 ± 0.017	0.337 ± 0.015	0.421 ± 0.020
	SUM-QE (MT) READ	**0.198**	**0.204** ± **0.005**	**0.384** ± **0.017**	**0.330** ± **0.015**	**0.409** ± **0.019**
REP	SUM-QE $\mathcal{Q}2$ (ST)	0.237	0.232 ± 0.005	0.363 ± 0.019	0.302 ± 0.016	0.385 ± 0.020
	SUM-QE REP (ST)	**0.202**	**0.200** ± **0.004**	0.503 ± 0.017	0.429 ± 0.015	0.531 ± 0.018
	SUM-QE (MT) REP	**0.202**	0.201 ± 0.004	**0.557** ± **0.015**	**0.478** ± **0.014**	**0.580** ± **0.017**

Table A.1.: Mean Absolute Error MAE on the *micro* level and *macro* MAE alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (SEM) between human (READ: *Readability*, REP: *Repetition*– Section 3.3) and automatic measure (Sum-QE trained on DUC and BioASQ datasets) at the document level using the validation data.

A.2 Additional results for content estimation

A.2.1 Sbert Cosine Similarity (CS)

Below we can see the performance of each sentence-bert transformer used in Section 5.2 to estimate the amount of shared information between the article and the summary calculating the cosine similarity between their embeddings. The results can be found in the Tables below by the name of the transformer that used as encoder, followed by (cs) (e.g., bert-base-nli-cls-token (cs)). We should mention that the name of each model indicates the way it was trained. For example the model bert-base-nli-cls-token uses the bert base model trained on the nli datasets Bowman et al., 2015; Williams et al., 2018 performing a mean pooling to all the output vectors. Reimers and Gurevych (2019) experimented with three pooling strategies: Using the output of the CLS-token, computing the mean of all the output vectors (MEANstrategy), and computing a max-over-time of the output vectors (MAX-strategy)

		NEWSROOM content estimations			
	Transformer	mae	ρ	τ	r
Informativeness	bert-base-nli-mean-tokens (cs)	0.270	0.711 ± 0.026	0.607 ± 0.027	0.773 ± 0.022
	bert-base-nli-max-tokens (cs)	0.352	0.727 ± 0.028	0.617 ± 0.029	**0.783** ± **0.023**
	bert-base-nli-cls-token (cs)	**0.279**	**0.735** ± **0.024**	**0.627** ± **0.026**	0.767 ± 0.023
	bert-large-nli-mean-tokens (cs)	0.286	0.710 ± 0.031	0.608 ± 0.031	0.776 ± 0.025
	bert-large-nli-cls-token (cs)	0.281	0.723 ± 0.032	0.620 ± 0.034	0.775 ± 0.025
Relevance	bert-base-nli-mean-tokens (cs)	**0.204**	0.625 ± 0.030	0.522 ± 0.029	0.780 ± 0.021
	bert-base-nli-max-tokens (cs)	0.282	0.633 ± 0.030	0.533 ± 0.030	**0.788** ± **0.021**
	bert-base-nli-cls-token (cs)	0.211	**0.640** ± **0.029**	**0.536** ± **0.028**	0.772 ± 0.022
	bert-large-nli-mean-tokens (cs)	0.218	0.616 ± 0.032	0.509 ± 0.031	0.778 ± 0.022
	bert-large-nli-cls-token (cs)	0.213	0.610 ± 0.034	0.504 ± 0.033	0.773 ± 0.022

Table A.2.: Mean Absolute Error mae alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (sem) between human measures (*Informativeness* and *Relevance*– Section 3.2) and the cosine similarity (cs) score between the summary's and document's embedding. The correlations are calculated at the document level.

		BioASQ (2018 val) content estimations				
		$Micro$	$Macro$			
	Method	mae	mae	ρ	τ	r
Information Precision	bert-base-nli-mean-tokens (cs)	**0.292**	**0.288 ± 0.007**	-0.135 ± 0.025	-0.120 ± 0.022	**0.113 ± 0.029**
	bert-base-nli-max-tokens (cs)	0.307	0.304 ± 0.008	-0.155 ± 0.025	-0.138 ± 0.022	0.100 ± 0.030
	bert-base-nli-cls-token (cs)	**0.292**	0.289 ± 0.007	**-0.118 ± 0.025**	**-0.103 ± 0.022**	0.109 ± 0.029
	bert-large-nli-mean-tokens (cs)	0.294	0.291 ± 0.007	-0.141 ± 0.025	-0.125 ± 0.022	0.089 ± 0.030
	bert-large-nli-cls-token (cs)	**0.292**	0.289 ± 0.007	-0.131 ± 0.025	-0.115 ± 0.022	0.096 ± 0.030
Information Recall	bert-base-nli-mean-tokens (cs)	0.207	0.205 ± 0.006	0.471 ± 0.018	0.414 ± 0.016	**0.586 ± 0.020**
	bert-base-nli-max-tokens (cs)	**0.205**	**0.202 ± 0.007**	**0.476 ± 0.018**	**0.419 ± 0.016**	0.585 ± 0.020
	bert-base-nli-cls-token (cs)	0.209	0.207 ± 0.006	0.454 ± 0.018	0.400 ± 0.016	0.582 ± 0.020
	bert-large-nli-mean-tokens (cs)	0.207	0.205 ± 0.006	0.466 ± 0.018	0.410 ± 0.017	0.583 ± 0.020
	bert-large-nli-cls-token (cs)	0.209	0.206 ± 0.006	0.455 ± 0.018	0.400 ± 0.017	0.579 ± 0.020

Table A.3.: Mean Absolute Error mae on the $micro$ level and $macro$ mae alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (sem) between human measures (*Information Precision* and *Information Recall*– Section 3.3) and the cosine similarity (cs) score between the summary's and document's embedding. The correlations are calculated at the document level.

		BioASQ (2019 Test) content estimations				
		$Micro$	$Macro$			
	Method	mae	mae	ρ	τ	r
Information Precision	bert-base-nli-mean-tokens (cs)	**0.208**	**0.201 ± 0.005**	-0.045 ± 0.035	-0.048 ± 0.033	**0.106 ± 0.038**
	bert-base-nli-max-tokens (cs)	0.207	0.198 ± 0.007	-0.063 ± 0.035	-0.066 ± 0.033	0.095 ± 0.038
	bert-base-nli-cls-token (cs)	**0.208**	0.200 ± 0.005	**-0.020 ± 0.035**	**-0.025 ± 0.033**	0.108 ± 0.038
	bert-large-nli-mean-tokens (cs)	0.209	0.201 ± 0.005	-0.029 ± 0.035	-0.034 ± 0.033	0.114 ± 0.038
	bert-large-nli-cls-token (cs)	**0.207**	0.199 ± 0.005	-0.021 ± 0.035	-0.026 ± 0.033	0.122 ± 0.038
Information Recall	bert-base-nli-mean-tokens (cs)	0.189	0.188 ± 0.006	0.601 ± 0.021	0.556 ± 0.020	**0.680 ± 0.021**
	bert-base-nli-max-tokens (cs)	**0.179**	**0.178 ± 0.007**	**0.611 ± 0.021**	**0.566 ± 0.020**	0.689 ± 0.020
	bert-base-nli-cls-token (cs)	0.191	0.190 ± 0.006	0.568 ± 0.023	0.527 ± 0.022	0.667 ± 0.022
	bert-large-nli-mean-tokens (cs)	0.187	0.186 ± 0.006	0.600 ± 0.021	0.556 ± 0.020	0.688 ± 0.021
	bert-large-nli-cls-token (cs)	0.189	0.188 ± 0.006	0.587 ± 0.022	0.544 ± 0.021	0.681 ± 0.022

Table A.4.: Mean Absolute Error mae alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (sem) between human (*Information Precision* and *Information Recall*– Section 3.3) measures and the cosine similarity (cs) between the summary's and document's embedding. The correlations are calculated at the document level. **BOLD** indicates the performance of the corresponding model that best performed on the validation data with respect the correlation level or the mae

A.2.2 Sbert LR

Below we can see the results from the experiments conducted where we trained a linear regressor (LR) to learn to weight the features of the embeddings produced by the sentence-bert transformers according to the measurement that we want to estimate. Below we can see the performance of each transformer that used alongside the prepossessing of the embeddings before passing through to the LR.

- (mt) indicates the Multi-Task and (st) the single task learning which is followed by the LR in order to be trained.
- (mult) indicates that the embeddings combined using element wise multiplication and (diff) the element-wise difference before passing through the LR.
- (norm) indicates that the embeddings were normalized using the L_2 norm before the combination of them.

For example, the bert-large-nli-mean-tokens (mt-diff-norm) indicates that we used the bert-large-nli-mean-tokens transformer as encoder for the summary and reference and the produced embeddings were normalized and combined using the element-wise difference before passing through the regressor.

	NEWSROOM *Informativeness* estimations			
Transformer	mae	ρ	τ	r
bert-base-nli-mean-tokens (st-mult-norm)	0.379	0.447 ± 0.054	0.401 ± 0.049	0.518 ± 0.057
bert-base-nli-mean-tokens (st-mult)	0.189	0.346 ± 0.044	0.266 ± 0.036	0.511 ± 0.040
bert-base-nli-mean-tokens (st-diff-norm)	0.363	0.179 ± 0.047	0.142 ± 0.041	0.270 ± 0.050
bert-base-nli-mean-tokens (st-diff)	0.231	0.275 ± 0.054	0.218 ± 0.045	0.402 ± 0.050
bert-base-nli-mean-tokens (mt-mult-norm)	0.357	**0.579 ± 0.027**	**0.520 ± 0.025**	**0.623 ± 0.032**
bert-base-nli-mean-tokens (mt-mult)	**0.185**	0.439 ± 0.043	0.351 ± 0.037	0.536 ± 0.046
bert-base-nli-mean-tokens (mt-diff-norm)	0.382	0.078 ± 0.055	0.058 ± 0.047	0.108 ± 0.056
bert-base-nli-mean-tokens (mt-diff)	0.223	0.253 ± 0.048	0.191 ± 0.040	0.409 ± 0.046
bert-base-nli-max-tokens (st-mult-norm)	0.402	0.308 ± 0.093	0.277 ± 0.083	0.344 ± 0.089
bert-base-nli-max-tokens (st-mult)	0.382	0.291 ± 0.042	0.259 ± 0.038	0.330 ± 0.040
bert-base-nli-max-tokens (st-diff-norm)	0.364	-0.001 ± 0.057	-0.006 ± 0.049	0.041 ± 0.058
bert-base-nli-max-tokens (st-diff)	0.285	0.067 ± 0.051	0.055 ± 0.041	0.132 ± 0.052
bert-base-nli-max-tokens (mt-mult-norm)	0.413	0.212 ± 0.091	0.192 ± 0.083	0.229 ± 0.087
bert-base-nli-max-tokens (mt-mult)	0.379	0.288 ± 0.058	0.253 ± 0.052	0.304 ± 0.059
bert-base-nli-max-tokens (mt-diff-norm)	0.401	-0.165 ± 0.054	-0.137 ± 0.046	-0.160 ± 0.056
bert-base-nli-max-tokens (mt-diff)	0.270	0.107 ± 0.053	0.081 ± 0.043	0.149 ± 0.055
bert-base-nli-cls-token (st-mult-norm)	0.372	0.450 ± 0.044	0.398 ± 0.039	0.521 ± 0.047
bert-base-nli-cls-token (st-mult)	0.194	0.424 ± 0.041	0.337 ± 0.036	0.526 ± 0.042
bert-base-nli-cls-token (st-diff-norm)	0.363	0.084 ± 0.053	0.065 ± 0.045	0.161 ± 0.055
bert-base-nli-cls-token (st-diff)	0.238	0.240 ± 0.051	0.193 ± 0.042	0.357 ± 0.051
bert-base-nli-cls-token (mt-mult-norm)	0.376	0.443 ± 0.054	0.396 ± 0.049	0.490 ± 0.063
bert-base-nli-cls-token (mt-mult)	0.206	0.339 ± 0.042	0.263 ± 0.036	0.440 ± 0.043
bert-base-nli-cls-token (mt-diff-norm)	0.368	0.047 ± 0.046	0.036 ± 0.038	0.122 ± 0.055
bert-base-nli-cls-token (mt-diff)	0.228	0.363 ± 0.043	0.290 ± 0.037	0.487 ± 0.040
bert-large-nli-mean-tokens (st-mult-norm)	0.375	0.470 ± 0.051	0.423 ± 0.046	0.547 ± 0.051
bert-large-nli-mean-tokens (st-mult)	0.253	0.284 ± 0.042	0.220 ± 0.035	0.445 ± 0.043
bert-large-nli-mean-tokens (st-diff-norm)	0.381	0.038 ± 0.060	0.026 ± 0.052	0.083 ± 0.059
bert-large-nli-mean-tokens (st-diff)	0.272	0.062 ± 0.059	0.046 ± 0.048	0.144 ± 0.061
bert-large-nli-mean-tokens (mt-mult-norm)	0.390	0.303 ± 0.062	0.274 ± 0.056	0.353 ± 0.069
bert-large-nli-mean-tokens (mt-mult)	0.284	0.298 ± 0.040	0.239 ± 0.034	0.426 ± 0.043
bert-large-nli-mean-tokens (mt-diff-norm)	0.354	0.132 ± 0.061	0.119 ± 0.052	0.177 ± 0.061
bert-large-nli-mean-tokens (mt-diff)	0.279	0.062 ± 0.059	0.047 ± 0.047	0.147 ± 0.057
bert-large-nli-cls-token (st-mult-norm)	0.382	0.369 ± 0.059	0.329 ± 0.053	0.435 ± 0.062
bert-large-nli-cls-token (st-mult)	0.276	0.350 ± 0.051	0.296 ± 0.044	0.496 ± 0.052
bert-large-nli-cls-token (st-diff-norm)	0.379	0.044 ± 0.054	0.039 ± 0.046	0.069 ± 0.055
bert-large-nli-cls-token (st-diff)	0.264	0.207 ± 0.052	0.168 ± 0.045	0.324 ± 0.050
bert-large-nli-cls-token (mt-mult-norm)	0.360	0.463 ± 0.051	0.410 ± 0.046	0.537 ± 0.053
bert-large-nli-cls-token (mt-mult)	0.294	0.330 ± 0.051	0.259 ± 0.044	0.463 ± 0.050
bert-large-nli-cls-token (mt-diff-norm)	0.363	0.090 ± 0.051	0.069 ± 0.044	0.119 ± 0.054
bert-large-nli-cls-token (mt-diff)	0.288	0.154 ± 0.055	0.102 ± 0.045	0.221 ± 0.057

Table A.5.: Mean Absolute Error mae alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (sem) between human (*Informativeness*–Section 3.2) scores and the scores produced by LR which was trained using the sentence-bert embeddings of the summary and the document. The correlations are calculated at the document level.

	NEWSROOM *Relevance* estimations			
Transformer	mae	ρ	τ	r
bert-base-nli-mean-tokens (st-mult-norm)	0.327	0.469 ± 0.062	0.430 ± 0.057	0.574 ± 0.070
bert-base-nli-mean-tokens (st-mult)	0.186	0.303 ± 0.048	0.236 ± 0.043	0.496 ± 0.047
bert-base-nli-mean-tokens (st-diff-norm)	0.359	-0.140 ± 0.044	-0.121 ± 0.038	-0.131 ± 0.041
bert-base-nli-mean-tokens (st-diff)	0.255	0.001 ± 0.053	-0.003 ± 0.043	0.080 ± 0.062
bert-base-nli-mean-tokens (mt-mult-norm)	0.333	0.367 ± 0.071	0.334 ± 0.064	0.396 ± 0.086
bert-base-nli-mean-tokens (mt-mult)	**0.179**	0.382 ± 0.047	0.289 ± 0.040	0.556 ± 0.044
bert-base-nli-mean-tokens (mt-diff-norm)	0.365	-0.166 ± 0.050	-0.138 ± 0.044	-0.157 ± 0.042
bert-base-nli-mean-tokens (mt-diff)	0.247	0.154 ± 0.052	0.108 ± 0.045	0.236 ± 0.056
bert-base-nli-max-tokens (st-mult-norm)	0.342	**0.598 ± 0.037**	**0.545 ± 0.036**	**0.726 ± 0.065**
bert-base-nli-max-tokens (st-mult)	0.401	0.329 ± 0.037	0.295 ± 0.034	0.397 ± 0.034
bert-base-nli-max-tokens (st-diff-norm)	0.330	-0.117 ± 0.054	-0.091 ± 0.046	-0.104 ± 0.056
bert-base-nli-max-tokens (st-diff)	0.321	-0.248 ± 0.049	-0.209 ± 0.044	-0.217 ± 0.048
bert-base-nli-max-tokens (mt-mult-norm)	0.343	0.332 ± 0.169	0.304 ± 0.153	0.295 ± 0.187
bert-base-nli-max-tokens (mt-mult)	0.327	0.361 ± 0.061	0.325 ± 0.055	0.364 ± 0.075
bert-base-nli-max-tokens (mt-diff-norm)	0.344	-0.084 ± 0.055	-0.077 ± 0.049	-0.022 ± 0.061
bert-base-nli-max-tokens (mt-diff)	0.296	0.037 ± 0.043	0.022 ± 0.036	0.138 ± 0.048
bert-base-nli-cls-token (st-mult-norm)	0.327	0.450 ± 0.056	0.407 ± 0.050	0.528 ± 0.071
bert-base-nli-cls-token (st-mult)	0.200	0.244 ± 0.047	0.199 ± 0.040	0.382 ± 0.053
bert-base-nli-cls-token (st-diff-norm)	0.343	-0.023 ± 0.062	-0.025 ± 0.054	-0.011 ± 0.060
bert-base-nli-cls-token (st-diff)	0.267	-0.055 ± 0.050	-0.045 ± 0.042	0.055 ± 0.055
bert-base-nli-cls-token (mt-mult-norm)	0.333	0.312 ± 0.058	0.283 ± 0.052	0.325 ± 0.072
bert-base-nli-cls-token (mt-mult)	0.210	0.184 ± 0.047	0.134 ± 0.039	0.331 ± 0.052
bert-base-nli-cls-token (mt-diff-norm)	0.315	0.249 ± 0.050	0.217 ± 0.044	0.304 ± 0.056
bert-base-nli-cls-token (mt-diff)	0.244	0.103 ± 0.058	0.072 ± 0.047	0.181 ± 0.060
bert-large-nli-mean-tokens (st-mult-norm)	0.336	0.361 ± 0.081	0.319 ± 0.074	0.368 ± 0.092
bert-large-nli-mean-tokens (st-mult)	0.289	0.134 ± 0.053	0.121 ± 0.047	0.202 ± 0.060
bert-large-nli-mean-tokens (st-diff-norm)	0.351	0.022 ± 0.064	0.015 ± 0.056	0.127 ± 0.067
bert-large-nli-mean-tokens (st-diff)	0.335	-0.242 ± 0.051	-0.207 ± 0.046	-0.217 ± 0.046
bert-large-nli-mean-tokens (mt-mult-norm)	0.326	0.398 ± 0.073	0.361 ± 0.066	0.472 ± 0.081
bert-large-nli-mean-tokens (mt-mult)	0.327	0.214 ± 0.054	0.191 ± 0.048	0.264 ± 0.054
bert-large-nli-mean-tokens (mt-diff-norm)	0.342	0.065 ± 0.053	0.053 ± 0.048	0.177 ± 0.056
bert-large-nli-mean-tokens (mt-diff)	0.355	-0.317 ± 0.046	-0.285 ± 0.041	-0.255 ± 0.041
bert-large-nli-cls-token (st-mult-norm)	0.332	0.299 ± 0.072	0.265 ± 0.065	0.356 ± 0.082
bert-large-nli-cls-token (st-mult)	0.295	0.264 ± 0.052	0.244 ± 0.046	0.273 ± 0.057
bert-large-nli-cls-token (st-diff-norm)	0.352	0.054 ± 0.055	0.043 ± 0.048	0.034 ± 0.057
bert-large-nli-cls-token (st-diff)	0.272	0.295 ± 0.042	0.222 ± 0.036	0.423 ± 0.046
bert-large-nli-cls-token (mt-mult-norm)	0.326	0.440 ± 0.054	0.401 ± 0.049	0.506 ± 0.069
bert-large-nli-cls-token (mt-mult)	0.306	0.114 ± 0.056	0.101 ± 0.049	0.227 ± 0.064
bert-large-nli-cls-token (mt-diff-norm)	0.332	0.109 ± 0.053	0.095 ± 0.047	0.152 ± 0.058
bert-large-nli-cls-token (mt-diff)	0.300	0.025 ± 0.055	0.007 ± 0.045	0.111 ± 0.060

Table A.6.: Mean Absolute Error mae alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (sem) between human (*Relevance*– Section 3.2) scores and the scores produced by LR which was trained using the sentence-bert embeddings of the summary and the document. The correlations are calculated at the document level.

	BioASQ (2018 Val) *Information Precision* estimations				
	Micro	*Macro*			
Method	MAE	MAE	ρ	τ	r
BERT-BASE-NLI-MEAN-TOKENS (ST-MULT-NORM)	0.365	0.349 ± 0.009	0.408 ± 0.046	0.395 ± 0.045	0.453 ± 0.050
BERT-BASE-NLI-MEAN-TOKENS (ST-MULT)	0.336	0.321 ± 0.008	0.286 ± 0.027	0.269 ± 0.025	0.380 ± 0.029
BERT-BASE-NLI-MEAN-TOKENS (ST-DIFF-NORM)	0.365	0.360 ± 0.007	0.047 ± 0.020	0.041 ± 0.018	0.031 ± 0.020
BERT-BASE-NLI-MEAN-TOKENS (ST-DIFF)	0.245	0.251 ± 0.005	0.218 ± 0.020	0.182 ± 0.018	0.302 ± 0.024
BERT-BASE-NLI-MEAN-TOKENS (MT-MULT-NORM)	0.359	0.343 ± 0.009	0.227 ± 0.034	0.220 ± 0.033	0.251 ± 0.037
BERT-BASE-NLI-MEAN-TOKENS (MT-MULT)	0.251	0.250 ± 0.006	0.254 ± 0.021	0.217 ± 0.018	0.321 ± 0.024
BERT-BASE-NLI-MEAN-TOKENS (MT-DIFF-NORM)	0.339	0.336 ± 0.006	0.108 ± 0.020	0.093 ± 0.018	0.104 ± 0.021
BERT-BASE-NLI-MEAN-TOKENS (MT-DIFF)	**0.236**	**0.239 ± 0.005**	0.297 ± 0.020	0.255 ± 0.018	0.357 ± 0.023
BERT-BASE-NLI-MAX-TOKENS (ST-MULT-NORM)	0.625	0.640 ± 0.009	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
BERT-BASE-NLI-MAX-TOKENS (ST-MULT)	0.353	0.336 ± 0.009	**0.502 ± 0.037**	**0.487 ± 0.037**	**0.565 ± 0.041**
BERT-BASE-NLI-MAX-TOKENS (ST-DIFF-NORM)	0.309	0.307 ± 0.006	0.167 ± 0.021	0.144 ± 0.018	0.197 ± 0.022
BERT-BASE-NLI-MAX-TOKENS (ST-DIFF)	0.269	0.270 ± 0.006	0.227 ± 0.022	0.197 ± 0.019	0.209 ± 0.024
BERT-BASE-NLI-MAX-TOKENS (MT-MULT-NORM)	0.625	0.640 ± 0.009	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
BERT-BASE-NLI-MAX-TOKENS (MT-MULT)	0.353	0.334 ± 0.009	0.499 ± 0.037	0.485 ± 0.036	0.563 ± 0.040
BERT-BASE-NLI-MAX-TOKENS (MT-DIFF-NORM)	0.311	0.312 ± 0.006	0.116 ± 0.020	0.099 ± 0.018	0.147 ± 0.023
BERT-BASE-NLI-MAX-TOKENS (MT-DIFF)	0.249	0.251 ± 0.005	0.237 ± 0.020	0.201 ± 0.018	0.303 ± 0.023
BERT-BASE-NLI-CLS-TOKEN (ST-MULT-NORM)	0.367	0.351 ± 0.009	0.366 ± 0.051	0.355 ± 0.050	0.425 ± 0.054
BERT-BASE-NLI-CLS-TOKEN (ST-MULT)	0.265	0.262 ± 0.006	0.277 ± 0.020	0.238 ± 0.018	0.357 ± 0.023
BERT-BASE-NLI-CLS-TOKEN (ST-DIFF-NORM)	0.382	0.374 ± 0.007	0.090 ± 0.020	0.083 ± 0.017	0.109 ± 0.020
BERT-BASE-NLI-CLS-TOKEN (ST-DIFF)	0.243	0.246 ± 0.005	0.247 ± 0.021	0.210 ± 0.018	0.292 ± 0.024
BERT-BASE-NLI-CLS-TOKEN (MT-MULT-NORM)	0.364	0.348 ± 0.009	0.219 ± 0.036	0.213 ± 0.034	0.253 ± 0.038
BERT-BASE-NLI-CLS-TOKEN (MT-MULT)	0.261	0.256 ± 0.006	0.278 ± 0.020	0.242 ± 0.018	0.369 ± 0.023
BERT-BASE-NLI-CLS-TOKEN (MT-DIFF-NORM)	0.351	0.343 ± 0.007	0.143 ± 0.019	0.126 ± 0.017	0.141 ± 0.019
BERT-BASE-NLI-CLS-TOKEN (MT-DIFF)	0.236	0.240 ± 0.005	0.261 ± 0.020	0.224 ± 0.018	0.311 ± 0.023
BERT-LARGE-NLI-MEAN-TOKENS (ST-MULT-NORM)	0.370	0.355 ± 0.009	0.409 ± 0.056	0.396 ± 0.054	0.443 ± 0.061
BERT-LARGE-NLI-MEAN-TOKENS (ST-MULT)	0.322	0.313 ± 0.008	0.269 ± 0.022	0.245 ± 0.021	0.310 ± 0.024
BERT-LARGE-NLI-MEAN-TOKENS (ST-DIFF-NORM)	0.383	0.376 ± 0.007	0.082 ± 0.020	0.071 ± 0.018	0.072 ± 0.020
BERT-LARGE-NLI-MEAN-TOKENS (ST-DIFF)	0.294	0.292 ± 0.006	0.218 ± 0.022	0.189 ± 0.020	0.211 ± 0.025
BERT-LARGE-NLI-MEAN-TOKENS (MT-MULT-NORM)	0.365	0.350 ± 0.009	0.145 ± 0.032	0.140 ± 0.030	0.157 ± 0.033
BERT-LARGE-NLI-MEAN-TOKENS (MT-MULT)	0.625	0.640 ± 0.009	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
BERT-LARGE-NLI-MEAN-TOKENS (MT-DIFF-NORM)	0.361	0.355 ± 0.007	0.076 ± 0.020	0.065 ± 0.018	0.064 ± 0.021
BERT-LARGE-NLI-MEAN-TOKENS (MT-DIFF)	0.253	0.256 ± 0.005	0.286 ± 0.019	0.243 ± 0.017	0.349 ± 0.022
BERT-LARGE-NLI-CLS-TOKEN (ST-MULT-NORM)	0.372	0.356 ± 0.009	0.404 ± 0.080	0.394 ± 0.079	0.408 ± 0.083
BERT-LARGE-NLI-CLS-TOKEN (ST-MULT)	0.325	0.319 ± 0.008	0.260 ± 0.023	0.239 ± 0.022	0.334 ± 0.025
BERT-LARGE-NLI-CLS-TOKEN (ST-DIFF-NORM)	0.375	0.374 ± 0.007	0.043 ± 0.021	0.037 ± 0.018	0.035 ± 0.021
BERT-LARGE-NLI-CLS-TOKEN (ST-DIFF)	0.254	0.254 ± 0.005	0.296 ± 0.019	0.257 ± 0.017	0.354 ± 0.022
BERT-LARGE-NLI-CLS-TOKEN (MT-MULT-NORM)	0.364	0.349 ± 0.009	0.195 ± 0.031	0.188 ± 0.030	0.200 ± 0.032
BERT-LARGE-NLI-CLS-TOKEN (MT-MULT)	0.317	0.308 ± 0.008	0.375 ± 0.021	0.346 ± 0.020	0.414 ± 0.023
BERT-LARGE-NLI-CLS-TOKEN (MT-DIFF-NORM)	0.352	0.349 ± 0.007	0.131 ± 0.021	0.115 ± 0.019	0.128 ± 0.022
BERT-LARGE-NLI-CLS-TOKEN (MT-DIFF)	0.249	0.249 ± 0.005	0.294 ± 0.020	0.254 ± 0.018	0.361 ± 0.023

Table A.7.: Mean Absolute Error MAE on the $micro$ level and $macro$ MAE alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (SEM) between human (*Information Precision*– Section 3.3) scores and the scores produced by LR which was trained using the SENTENCE-BERT embeddings of the summary and the document. The correlations are calculated at the document level.

	BioASQ (2019 Test) *Information Precision* estimations				
	Micro	*Macro*			
Method	MAE	MAE	ρ	τ	r
BERT-BASE-NLI-MEAN-TOKENS (ST-MULT-NORM)	0.203	0.195 ± 0.008	0.297 ± 0.057	0.292 ± 0.057	0.331 ± 0.059
BERT-BASE-NLI-MEAN-TOKENS (ST-MULT)	0.194	0.188 ± 0.007	0.316 ± 0.036	0.303 ± 0.035	0.351 ± 0.037
BERT-BASE-NLI-MEAN-TOKENS (ST-DIFF-NORM)	0.313	0.307 ± 0.009	0.048 ± 0.032	0.046 ± 0.031	0.062 ± 0.032
BERT-BASE-NLI-MEAN-TOKENS (ST-DIFF)	0.287	0.283 ± 0.006	0.149 ± 0.032	0.140 ± 0.030	0.223 ± 0.034
BERT-BASE-NLI-MEAN-TOKENS (MT-MULT-NORM)	0.205	0.197 ± 0.008	0.270 ± 0.046	0.268 ± 0.046	0.279 ± 0.048
BERT-BASE-NLI-MEAN-TOKENS (MT-MULT)	0.265	0.258 ± 0.007	0.238 ± 0.033	0.220 ± 0.031	0.249 ± 0.034
BERT-BASE-NLI-MEAN-TOKENS (MT-DIFF-NORM)	0.313	0.305 ± 0.008	0.100 ± 0.031	0.095 ± 0.029	0.110 ± 0.032
BERT-BASE-NLI-MEAN-TOKENS (MT-DIFF)	**0.268**	**0.262** ± **0.006**	0.197 ± 0.032	0.186 ± 0.030	0.255 ± 0.034
BERT-BASE-NLI-MAX-TOKENS (ST-MULT-NORM)	0.796	0.806 ± 0.008	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
BERT-BASE-NLI-MAX-TOKENS (ST-MULT)	0.202	0.193 ± 0.008	**0.323** ± **0.041**	**0.318** ± **0.040**	**0.377** ± **0.044**
BERT-BASE-NLI-MAX-TOKENS (ST-DIFF-NORM)	0.332	0.331 ± 0.008	0.150 ± 0.033	0.142 ± 0.031	0.180 ± 0.034
BERT-BASE-NLI-MAX-TOKENS (ST-DIFF)	0.248	0.239 ± 0.006	0.131 ± 0.033	0.124 ± 0.031	0.110 ± 0.033
BERT-BASE-NLI-MAX-TOKENS (MT-MULT-NORM)	0.796	0.806 ± 0.008	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
BERT-BASE-NLI-MAX-TOKENS (MT-MULT)	0.209	0.199 ± 0.008	0.326 ± 0.041	0.321 ± 0.040	0.380 ± 0.043
BERT-BASE-NLI-MAX-TOKENS (MT-DIFF-NORM)	0.328	0.323 ± 0.008	0.153 ± 0.032	0.146 ± 0.030	0.189 ± 0.032
BERT-BASE-NLI-MAX-TOKENS (MT-DIFF)	0.275	0.273 ± 0.006	0.197 ± 0.031	0.182 ± 0.029	0.259 ± 0.033
BERT-BASE-NLI-CLS-TOKEN (ST-MULT-NORM)	0.199	0.190 ± 0.008	0.303 ± 0.055	0.297 ± 0.054	0.356 ± 0.061
BERT-BASE-NLI-CLS-TOKEN (ST-MULT)	0.243	0.241 ± 0.008	0.194 ± 0.034	0.181 ± 0.032	0.261 ± 0.035
BERT-BASE-NLI-CLS-TOKEN (ST-DIFF-NORM)	0.381	0.386 ± 0.010	0.049 ± 0.032	0.044 ± 0.031	0.077 ± 0.033
BERT-BASE-NLI-CLS-TOKEN (ST-DIFF)	0.260	0.253 ± 0.006	0.189 ± 0.032	0.177 ± 0.030	0.223 ± 0.033
BERT-BASE-NLI-CLS-TOKEN (MT-MULT-NORM)	0.204	0.195 ± 0.008	0.215 ± 0.047	0.210 ± 0.046	0.247 ± 0.050
BERT-BASE-NLI-CLS-TOKEN (MT-MULT)	0.227	0.223 ± 0.007	0.214 ± 0.033	0.198 ± 0.031	0.277 ± 0.035
BERT-BASE-NLI-CLS-TOKEN (MT-DIFF-NORM)	0.341	0.343 ± 0.009	0.062 ± 0.032	0.060 ± 0.031	0.089 ± 0.033
BERT-BASE-NLI-CLS-TOKEN (MT-DIFF)	0.260	0.254 ± 0.006	0.212 ± 0.031	0.200 ± 0.029	0.265 ± 0.032
BERT-LARGE-NLI-MEAN-TOKENS (ST-MULT-NORM)	0.204	0.193 ± 0.008	0.353 ± 0.064	0.346 ± 0.063	0.385 ± 0.068
BERT-LARGE-NLI-MEAN-TOKENS (ST-MULT)	0.243	0.237 ± 0.009	0.261 ± 0.035	0.250 ± 0.033	0.272 ± 0.035
BERT-LARGE-NLI-MEAN-TOKENS (ST-DIFF-NORM)	0.359	0.359 ± 0.010	0.149 ± 0.032	0.142 ± 0.030	0.161 ± 0.032
BERT-LARGE-NLI-MEAN-TOKENS (ST-DIFF)	0.246	0.235 ± 0.006	0.133 ± 0.034	0.126 ± 0.032	0.112 ± 0.034
BERT-LARGE-NLI-MEAN-TOKENS (MT-MULT-NORM)	0.208	0.197 ± 0.008	0.144 ± 0.047	0.138 ± 0.046	0.156 ± 0.049
BERT-LARGE-NLI-MEAN-TOKENS (MT-MULT)	0.796	0.806 ± 0.008	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
BERT-LARGE-NLI-MEAN-TOKENS (MT-DIFF-NORM)	0.324	0.315 ± 0.008	0.138 ± 0.031	0.133 ± 0.029	0.132 ± 0.030
BERT-LARGE-NLI-MEAN-TOKENS (MT-DIFF)	0.278	0.276 ± 0.006	0.175 ± 0.031	0.165 ± 0.029	0.252 ± 0.033
BERT-LARGE-NLI-CLS-TOKEN (ST-MULT-NORM)	0.204	0.194 ± 0.008	0.223 ± 0.075	0.217 ± 0.074	0.270 ± 0.081
BERT-LARGE-NLI-CLS-TOKEN (ST-MULT)	0.238	0.240 ± 0.009	0.306 ± 0.033	0.290 ± 0.032	0.338 ± 0.035
BERT-LARGE-NLI-CLS-TOKEN (ST-DIFF-NORM)	0.343	0.338 ± 0.009	-0.009 ± 0.032	-0.005 ±0.031	-0.020 ± 0.033
BERT-LARGE-NLI-CLS-TOKEN (ST-DIFF)	0.263	0.255 ± 0.006	0.215 ± 0.031	0.200 ± 0.030	0.279 ± 0.032
BERT-LARGE-NLI-CLS-TOKEN (MT-MULT-NORM)	0.210	0.200 ± 0.008	0.160 ± 0.045	0.155 ± 0.044	0.174 ± 0.046
BERT-LARGE-NLI-CLS-TOKEN (MT-MULT)	0.232	0.227 ± 0.009	0.338 ± 0.035	0.325 ± 0.034	0.366 ± 0.036
BERT-LARGE-NLI-CLS-TOKEN (MT-DIFF-NORM)	0.326	0.317 ± 0.009	0.054 ± 0.032	0.051 ± 0.031	0.082 ± 0.033
BERT-LARGE-NLI-CLS-TOKEN (MT-DIFF)	0.277	0.273 ± 0.006	0.254 ± 0.030	0.236 ± 0.028	0.306 ± 0.031

Table A.8.: Mean Absolute Error MAE on the *micro* level and *macro* MAE alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (SEM) between human (*Information Precision*– Section 3.3) scores and the scores produced by LR which was trained using the SENTENCE-BERT embeddings of the summary and the document. The correlations are calculated at the document level. **BOLD** indicates the performance of the corresponding model that best performed on the validation data with respect the correlation level or the MAE

	BioASQ (2018 Val) *Information Recall* estimations				
	Micro	*Macro*			
Method	MAE	MAE	ρ	τ	r
BERT-BASE-NLI-MEAN-TOKENS (ST-MULT-NORM)	0.182	0.180 ± 0.008	0.608 ± 0.029	0.593 ± 0.029	0.695 ± 0.031
BERT-BASE-NLI-MEAN-TOKENS (ST-MULT)	0.168	0.163 ± 0.007	0.551 ± 0.020	0.520 ± 0.019	0.623 ± 0.021
BERT-BASE-NLI-MEAN-TOKENS (ST-DIFF-NORM)	0.364	0.357 ± 0.009	0.203 ± 0.021	0.184 ± 0.019	0.225 ± 0.022
BERT-BASE-NLI-MEAN-TOKENS (ST-DIFF)	0.213	0.211 ± 0.005	0.271 ± 0.021	0.238 ± 0.019	0.391 ± 0.024
BERT-BASE-NLI-MEAN-TOKENS (MT-MULT-NORM)	0.179	0.177 ± 0.008	0.616 ± 0.026	0.599 ± 0.026	0.689 ± 0.028
BERT-BASE-NLI-MEAN-TOKENS (MT-MULT)	0.168	0.164 ± 0.006	0.538 ± 0.018	0.503 ± 0.018	0.632 ± 0.020
BERT-BASE-NLI-MEAN-TOKENS (MT-DIFF-NORM)	0.296	0.299 ± 0.009	0.221 ± 0.022	0.206 ± 0.021	0.288 ± 0.024
BERT-BASE-NLI-MEAN-TOKENS (MT-DIFF)	0.210	0.208 ± 0.005	0.301 ± 0.020	0.263 ± 0.018	0.444 ± 0.023
BERT-BASE-NLI-MAX-TOKENS (ST-MULT-NORM)	0.182	0.180 ± 0.008	**0.676** ± **0.026**	**0.660** ± **0.026**	**0.751** ± **0.027**
BERT-BASE-NLI-MAX-TOKENS (ST-MULT)	0.799	0.802 ± 0.008	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
BERT-BASE-NLI-MAX-TOKENS (ST-DIFF-NORM)	0.331	0.327 ± 0.008	0.255 ± 0.020	0.223 ± 0.018	0.319 ± 0.021
BERT-BASE-NLI-MAX-TOKENS (ST-DIFF)	0.230	0.222 ± 0.006	0.370 ± 0.019	0.326 ± 0.017	0.475 ± 0.021
BERT-BASE-NLI-MAX-TOKENS (MT-MULT-NORM)	0.183	0.180 ± 0.008	0.652 ± 0.026	0.636 ± 0.027	0.739 ± 0.028
BERT-BASE-NLI-MAX-TOKENS (MT-MULT)	0.169	0.165 ± 0.007	0.658 ± 0.022	0.642 ± 0.022	0.742 ± 0.023
BERT-BASE-NLI-MAX-TOKENS (MT-DIFF-NORM)	0.300	0.303 ± 0.008	-0.019 ± 0.023	-0.014 ± 0.021	0.031 ± 0.024
BERT-BASE-NLI-MAX-TOKENS (MT-DIFF)	0.192	0.187 ± 0.006	0.259 ± 0.022	0.233 ± 0.020	0.381 ± 0.025
BERT-BASE-NLI-CLS-TOKEN (ST-MULT-NORM)	0.181	0.179 ± 0.008	0.607 ± 0.027	0.591 ± 0.027	0.688 ± 0.029
BERT-BASE-NLI-CLS-TOKEN (ST-MULT)	0.163	0.159 ± 0.007	0.586 ± 0.019	0.560 ± 0.019	0.652 ± 0.020
BERT-BASE-NLI-CLS-TOKEN (ST-DIFF-NORM)	0.345	0.337 ± 0.009	0.117 ± 0.021	0.105 ± 0.020	0.136 ± 0.022
BERT-BASE-NLI-CLS-TOKEN (ST-DIFF)	0.231	0.226 ± 0.005	0.347 ± 0.019	0.300 ± 0.017	0.469 ± 0.022
BERT-BASE-NLI-CLS-TOKEN (MT-MULT-NORM)	0.179	0.176 ± 0.007	0.623 ± 0.026	0.607 ± 0.026	0.699 ± 0.028
BERT-BASE-NLI-CLS-TOKEN (MT-MULT)	0.162	0.157 ± 0.007	0.554 ± 0.021	0.532 ± 0.020	0.622 ± 0.022
BERT-BASE-NLI-CLS-TOKEN (MT-DIFF-NORM)	0.301	0.302 ± 0.008	0.112 ± 0.022	0.104 ± 0.021	0.159 ± 0.023
BERT-BASE-NLI-CLS-TOKEN (MT-DIFF)	0.199	0.199 ± 0.005	0.315 ± 0.019	0.275 ± 0.017	0.451 ± 0.022
BERT-LARGE-NLI-MEAN-TOKENS (ST-MULT-NORM)	0.180	0.175 ± 0.007	0.566 ± 0.028	0.552 ± 0.028	0.626 ± 0.030
BERT-LARGE-NLI-MEAN-TOKENS (ST-MULT)	0.202	0.195 ± 0.008	0.554 ± 0.021	0.530 ± 0.020	0.589 ± 0.022
BERT-LARGE-NLI-MEAN-TOKENS (ST-DIFF-NORM)	0.398	0.386 ± 0.010	0.214 ± 0.019	0.192 ± 0.018	0.238 ± 0.020
BERT-LARGE-NLI-MEAN-TOKENS (ST-DIFF)	0.196	0.191 ± 0.006	0.288 ± 0.021	0.258 ± 0.019	0.415 ± 0.023
BERT-LARGE-NLI-MEAN-TOKENS (MT-MULT-NORM)	0.180	0.174 ± 0.007	0.567 ± 0.027	0.551 ± 0.027	0.619 ± 0.030
BERT-LARGE-NLI-MEAN-TOKENS (MT-MULT)	0.163	0.157 ± 0.007	0.618 ± 0.021	0.599 ± 0.021	0.677 ± 0.022
BERT-LARGE-NLI-MEAN-TOKENS (MT-DIFF-NORM)	0.294	0.297 ± 0.009	0.163 ± 0.023	0.151 ± 0.021	0.213 ± 0.024
BERT-LARGE-NLI-MEAN-TOKENS (MT-DIFF)	0.184	0.181 ± 0.006	0.356 ± 0.021	0.326 ± 0.019	0.473 ± 0.023
BERT-LARGE-NLI-CLS-TOKEN (ST-MULT-NORM)	0.176	0.171 ± 0.007	0.542 ± 0.025	0.527 ± 0.025	0.596 ± 0.027
BERT-LARGE-NLI-CLS-TOKEN (ST-MULT)	0.163	0.157 ± 0.007	0.637 ± 0.020	0.621 ± 0.020	0.709 ± 0.022
BERT-LARGE-NLI-CLS-TOKEN (ST-DIFF-NORM)	0.350	0.345 ± 0.009	0.210 ± 0.021	0.192 ± 0.019	0.236 ± 0.021
BERT-LARGE-NLI-CLS-TOKEN (ST-DIFF)	0.210	0.207 ± 0.007	0.381 ± 0.020	0.344 ± 0.019	0.470 ± 0.022
BERT-LARGE-NLI-CLS-TOKEN (MT-MULT-NORM)	0.178	0.173 ± 0.007	0.584 ± 0.026	0.570 ± 0.026	0.649 ± 0.028
BERT-LARGE-NLI-CLS-TOKEN (MT-MULT)	**0.159**	**0.155** ± **0.007**	0.607 ± 0.021	0.590 ± 0.021	0.670 ± 0.022
BERT-LARGE-NLI-CLS-TOKEN (MT-DIFF-NORM)	0.280	0.287 ± 0.008	0.005 ± 0.025	0.005 ± 0.023	0.033 ± 0.025
BERT-LARGE-NLI-CLS-TOKEN (MT-DIFF)	0.180	0.176 ± 0.007	0.337 ± 0.021	0.314 ± 0.020	0.446 ± 0.023

Table A.9.: Mean Absolute Error MAE on the $micro$ level and $macro$ MAE alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (SEM) between human (*Information Precision*– Section 3.3) scores and the scores produced by LR which was trained using the SENTENCE-BERT embeddings of the summary and the document. The correlations are calculated at the document level.

	BioASQ (2019 Test) *Information Recall* estimations				
	Micro	*Macro*			
Method	mae	mae	ρ	τ	r
bert-base-nli-mean-tokens (st-mult-norm)	0.165	0.167 ± 0.009	0.524 ± 0.030	0.516 ± 0.030	0.561 ± 0.032
bert-base-nli-mean-tokens (st-mult)	0.153	0.153 ± 0.007	0.660 ± 0.022	0.637 ± 0.021	0.671 ± 0.022
bert-base-nli-mean-tokens (st-diff-norm)	0.320	0.325 ± 0.010	0.211 ± 0.033	0.204 ± 0.031	0.265 ± 0.032
bert-base-nli-mean-tokens (st-diff)	0.212	0.209 ± 0.007	0.366 ± 0.030	0.338 ± 0.028	0.447 ± 0.030
bert-base-nli-mean-tokens (mt-mult-norm)	0.164	0.165 ± 0.008	0.506 ± 0.031	0.499 ± 0.031	0.539 ± 0.032
bert-base-nli-mean-tokens (mt-mult)	0.163	0.163 ± 0.007	0.624 ± 0.023	0.596 ± 0.023	0.669 ± 0.023
bert-base-nli-mean-tokens (mt-diff-norm)	0.314	0.311 ± 0.010	0.296 ± 0.032	0.284 ± 0.031	0.359 ± 0.032
bert-base-nli-mean-tokens (mt-diff)	0.200	0.197 ± 0.006	0.434 ± 0.028	0.403 ± 0.026	0.541 ± 0.028
bert-base-nli-max-tokens (st-mult-norm)	0.167	0.166 ± 0.009	**0.562 ± 0.031**	**0.553 ± 0.031**	**0.585 ± 0.032**
bert-base-nli-max-tokens (st-mult)	0.825	0.825 ± 0.009	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
bert-base-nli-max-tokens (st-diff-norm)	0.326	0.327 ± 0.009	0.341 ± 0.029	0.319 ± 0.028	0.411 ± 0.029
bert-base-nli-max-tokens (st-diff)	0.234	0.235 ± 0.007	0.487 ± 0.026	0.453 ± 0.025	0.587 ± 0.025
bert-base-nli-max-tokens (mt-mult-norm)	0.165	0.164 ± 0.009	0.555 ± 0.031	0.546 ± 0.031	0.580 ± 0.032
bert-base-nli-max-tokens (mt-mult)	0.164	0.166 ± 0.009	0.528 ± 0.029	0.520 ± 0.029	0.565 ± 0.030
bert-base-nli-max-tokens (mt-diff-norm)	0.290	0.284 ± 0.009	-0.094 ± 0.034	-0.091 ± 0.032	-0.047 ± 0.036
bert-base-nli-max-tokens (mt-diff)	0.188	0.187 ± 0.007	0.295 ± 0.032	0.276 ± 0.030	0.416 ± 0.032
bert-base-nli-cls-token (st-mult-norm)	0.165	0.164 ± 0.008	0.514 ± 0.033	0.505 ± 0.033	0.533 ± 0.034
bert-base-nli-cls-token (st-mult)	0.158	0.160 ± 0.008	0.629 ± 0.024	0.610 ± 0.024	0.643 ± 0.024
bert-base-nli-cls-token (st-diff-norm)	0.327	0.329 ± 0.010	0.162 ± 0.032	0.154 ± 0.031	0.200 ± 0.032
bert-base-nli-cls-token (st-diff)	0.229	0.227 ± 0.007	0.453 ± 0.027	0.418 ± 0.026	0.553 ± 0.027
bert-base-nli-cls-token (mt-mult-norm)	0.165	0.163 ± 0.008	0.526 ± 0.031	0.517 ± 0.031	0.544 ± 0.032
bert-base-nli-cls-token (mt-mult)	0.157	0.159 ± 0.008	0.652 ± 0.023	0.632 ± 0.023	0.666 ± 0.023
bert-base-nli-cls-token (mt-diff-norm)	0.291	0.288 ± 0.010	0.145 ± 0.037	0.140 ± 0.035	0.199 ± 0.037
bert-base-nli-cls-token (mt-diff)	0.203	0.201 ± 0.006	0.387 ± 0.029	0.361 ± 0.028	0.510 ± 0.028
bert-large-nli-mean-tokens (st-mult-norm)	0.166	0.166 ± 0.008	0.563 ± 0.032	0.553 ± 0.032	0.588 ± 0.032
bert-large-nli-mean-tokens (st-mult)	0.194	0.194 ± 0.009	0.631 ± 0.025	0.614 ± 0.024	0.637 ± 0.025
bert-large-nli-mean-tokens (st-diff-norm)	0.372	0.375 ± 0.012	0.387 ± 0.029	0.367 ± 0.028	0.406 ± 0.028
bert-large-nli-mean-tokens (st-diff)	0.179	0.179 ± 0.007	0.371 ± 0.031	0.350 ± 0.030	0.496 ± 0.030
bert-large-nli-mean-tokens (mt-mult-norm)	0.165	0.165 ± 0.008	0.541 ± 0.032	0.531 ± 0.032	0.562 ± 0.033
bert-large-nli-mean-tokens (mt-mult)	0.157	0.160 ± 0.008	0.672 ± 0.021	0.657 ± 0.021	0.687 ± 0.021
bert-large-nli-mean-tokens (mt-diff-norm)	0.308	0.308 ± 0.010	0.267 ± 0.032	0.255 ± 0.031	0.313 ± 0.032
bert-large-nli-mean-tokens (mt-diff)	0.179	0.182 ± 0.007	0.432 ± 0.029	0.410 ± 0.028	0.522 ± 0.028
bert-large-nli-cls-token (st-mult-norm)	0.168	0.170 ± 0.008	0.552 ± 0.029	0.543 ± 0.029	0.549 ± 0.030
bert-large-nli-cls-token (st-mult)	0.157	0.158 ± 0.008	0.630 ± 0.023	0.619 ± 0.023	0.646 ± 0.024
bert-large-nli-cls-token (st-diff-norm)	0.324	0.321 ± 0.011	0.317 ± 0.031	0.303 ± 0.030	0.363 ± 0.030
bert-large-nli-cls-token (st-diff)	0.212	0.210 ± 0.008	0.538 ± 0.026	0.512 ± 0.025	0.576 ± 0.025
bert-large-nli-cls-token (mt-mult-norm)	0.166	0.168 ± 0.008	0.541 ± 0.029	0.532 ± 0.029	0.567 ± 0.030
bert-large-nli-cls-token (mt-mult)	**0.161**	**0.163 ± 0.008**	0.649 ± 0.023	0.635 ± 0.023	0.660 ± 0.023
bert-large-nli-cls-token (mt-diff-norm)	0.278	0.275 ± 0.009	0.026 ± 0.039	0.024 ± 0.038	0.066 ± 0.039
bert-large-nli-cls-token (mt-diff)	0.169	0.169 ± 0.007	0.451 ± 0.028	0.431 ± 0.027	0.535 ± 0.027

Table A.10.: Mean Absolute Error mae on the *micro* level and *macro* mae alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (sem) between human (*Information Precision*– Section 3.3) scores and the scores produced by LR which was trained using the sentence-bert embeddings of the summary and the document. The correlations are calculated at the document level. **BOLD** indicates the performance of the corresponding model that best performed on the validation data with respect the correlation level or the mae.

A.2.3 SUPERT & Alt. SUPERT

Below we can see the results from the experiments conducted where we used the SUPERT model for our content evaluation. Specifically, we experimented with the same pipeline published by Gao et al. (2020) but in some of our experiments we changed the SENTENCE-BERT encoder and we disabled the preudo-ref mechanism in order to obtain whether giving other type of documents can achieve better content estimation. In the tables below, each name indicates the transformer that we used as encoder alongside the document type and the output metric.

- (MECH) indicates the we used the provided mechanism to build the pseudo-reference.
- (REF) indicates the we used the real reference.
- (SD) indicates that we used the concatenation of all the source documents as pseudo-reference.
- (SNIP) indicates that we used the concatenation of the snippets as pseudo-reference (only in BioASQ dataset).

For example, BERT-BASE-NLI-MAX-TOKENS (MECH-REC) indicates that we used the BERT-BASE-NLI-MAX-TOKENS model as encoder comparing the candidate summary with the pseudo-ref summary produced by the published mechanism MECH using the *Information Recall* (rec) measure as the estimation. The SUPERT model uses the BERT-LARGE-NLI-STSB-MEAN-TOKENS as default encoder and the pseudo-ref summary produced by the published mechanism in order to compare it with the candidate summary. Hence, the names BERT-LARGE-NLI-STSB-MEAN-TOKENS (MECH-REC), BERT-LARGE-NLI-STSB-MEAN-TOKENS (MECH-PREC) and BERT-LARGE-NLI-STSB-MEAN-TOKENS (MECH-F1) correspond to the original SUPERT model.

	NEWSROOM *Informativeness* estimations			
Transformer	mae	ρ	τ	r
bert-base-nli-mean-tokens (ref-rec)	0.188	0.335 ± 0.038	0.293 ± 0.031	0.387 ± 0.045
bert-base-nli-mean-tokens (ref-f1)	0.199	0.229 ± 0.039	0.194 ± 0.033	0.272 ± 0.051
bert-base-nli-mean-tokens (sd-rec)	0.122	0.715 ± 0.027	0.612 ± 0.027	0.776 ± 0.024
bert-base-nli-mean-tokens (sd-f1)	0.125	0.701 ± 0.024	0.593 ± 0.025	0.779 ± 0.024
bert-base-nli-mean-tokens (mech-rec)	0.120	0.694 ± 0.028	0.588 ± 0.029	0.774 ± 0.023
bert-base-nli-mean-tokens (mech-f1)	0.127	0.682 ± 0.029	0.577 ± 0.028	0.770 ± 0.024
bert-base-nli-max-tokens (ref-rec)	0.179	0.347 ± 0.037	0.301 ± 0.033	0.438 ± 0.041
bert-base-nli-max-tokens (ref-f1)	0.190	0.255 ± 0.035	0.217 ± 0.031	0.321 ± 0.048
bert-base-nli-max-tokens (sd-rec)	0.119	**0.726 ± 0.027**	**0.625 ± 0.029**	0.778 ± 0.024
bert-base-nli-max-tokens (sd-f1)	0.136	0.699 ± 0.025	0.591 ± 0.027	0.781 ± 0.023
bert-base-nli-max-tokens (mech-rec)	**0.118**	0.711 ± 0.029	0.611 ± 0.030	0.777 ± 0.023
bert-base-nli-max-tokens (mech-f1)	0.138	0.686 ± 0.030	0.584 ± 0.030	0.772 ± 0.024
bert-base-nli-cls-token (ref-rec)	0.178	0.316 ± 0.039	0.271 ± 0.033	0.390 ± 0.044
bert-base-nli-cls-token (ref-f1)	0.189	0.228 ± 0.040	0.202 ± 0.035	0.257 ± 0.052
bert-base-nli-cls-token (sd-rec)	0.122	0.722 ± 0.025	0.617 ± 0.027	0.775 ± 0.024
bert-base-nli-cls-token (sd-f1)	0.137	0.701 ± 0.026	0.601 ± 0.027	0.778 ± 0.023
bert-base-nli-cls-token (mech-rec)	0.120	0.718 ± 0.025	0.611 ± 0.027	0.773 ± 0.023
bert-base-nli-cls-token (mech-f1)	0.138	0.685 ± 0.030	0.593 ± 0.029	0.765 ± 0.024
bert-large-nli-mean-tokens (ref-rec)	0.177	0.312 ± 0.042	0.266 ± 0.036	0.417 ± 0.045
bert-large-nli-mean-tokens (ref-f1)	0.184	0.229 ± 0.040	0.189 ± 0.035	0.325 ± 0.048
bert-large-nli-mean-tokens (sd-rec)	0.124	0.703 ± 0.027	0.596 ± 0.028	0.766 ± 0.025
bert-large-nli-mean-tokens (sd-f1)	0.151	0.649 ± 0.027	0.547 ± 0.026	0.768 ± 0.024
bert-large-nli-mean-tokens (mech-rec)	0.124	0.674 ± 0.029	0.564 ± 0.030	0.765 ± 0.024
bert-large-nli-mean-tokens (mech-f1)	0.151	0.642 ± 0.032	0.539 ± 0.031	0.758 ± 0.025
bert-large-nli-cls-token (ref-rec)	0.184	0.312 ± 0.040	0.271 ± 0.034	0.374 ± 0.046
bert-large-nli-cls-token (ref-f1)	0.189	0.223 ± 0.042	0.183 ± 0.036	0.274 ± 0.052
bert-large-nli-cls-token (sd-rec)	0.132	0.700 ± 0.024	0.590 ± 0.025	0.767 ± 0.024
bert-large-nli-cls-token (sd-f1)	0.168	0.664 ± 0.027	0.561 ± 0.027	0.765 ± 0.024
bert-large-nli-cls-token (mech-rec)	0.133	0.702 ± 0.024	0.591 ± 0.026	0.763 ± 0.024
bert-large-nli-cls-token (mech-f1)	0.168	0.658 ± 0.030	0.557 ± 0.029	0.752 ± 0.025
bert-large-nli-stsb-mean-tokens (ref-rec)	0.251	0.283 ± 0.041	0.235 ± 0.035	0.309 ± 0.048
bert-large-nli-stsb-mean-tokens (ref-f1)	0.268	0.197 ± 0.042	0.164 ± 0.037	0.200 ± 0.054
bert-large-nli-stsb-mean-tokens (sd-rec)	0.205	0.723 ± 0.027	0.623 ± 0.027	**0.784 ± 0.024**
bert-large-nli-stsb-mean-tokens (sd-f1)	0.136	0.688 ± 0.025	0.580 ± 0.026	0.778 ± 0.023
bert-large-nli-stsb-mean-tokens (mech-rec)	0.194	0.715 ± 0.027	0.611 ± 0.029	0.779 ± 0.022
bert-large-nli-stsb-mean-tokens (mech-f1)	0.135	0.684 ± 0.028	0.574 ± 0.028	0.762 ± 0.024

Table A.11.: Mean Absolute Error mae alongside the Spearman's ρ, Kendall's τ nad Pearson's r correlations ± Standard Error of the Mean (sem) between human (*Informativeness*– Section 3.2) scores and the scores predicted by the supert and the alternative versions, as described in Section 5.2. The correlations are calculated at the document level.

	NEWSROOM *Relevance* estimations			
Transformer	mae	ρ	τ	r
bert-base-nli-mean-tokens (ref-prec)	0.228	0.159 ± 0.047	0.132 ± 0.040	0.181 ± 0.057
bert-base-nli-mean-tokens (ref-f1)	0.210	0.209 ± 0.044	0.170 ± 0.038	0.276 ± 0.054
bert-base-nli-mean-tokens (sd-prec)	0.169	0.577 ± 0.036	0.476 ± 0.034	0.772 ± 0.025
bert-base-nli-mean-tokens (sd-f1)	**0.106**	**0.636 ± 0.027**	**0.526 ± 0.027**	**0.794 ± 0.022**
bert-base-nli-mean-tokens (mech-prec)	0.160	0.552 ± 0.036	0.459 ± 0.033	0.737 ± 0.027
bert-base-nli-mean-tokens (mech-f1)	0.108	0.599 ± 0.029	0.497 ± 0.028	0.777 ± 0.022
bert-base-nli-max-tokens (ref-prec)	0.216	0.184 ± 0.047	0.152 ± 0.041	0.218 ± 0.057
bert-base-nli-max-tokens (ref-f1)	0.197	0.225 ± 0.042	0.178 ± 0.037	0.328 ± 0.051
bert-base-nli-max-tokens (sd-prec)	0.181	0.581 ± 0.035	0.483 ± 0.034	0.771 ± 0.025
bert-base-nli-max-tokens (sd-f1)	0.108	0.629 ± 0.031	0.515 ± 0.030	0.792 ± 0.022
bert-base-nli-max-tokens (mech-prec)	0.173	0.547 ± 0.037	0.460 ± 0.035	0.742 ± 0.026
bert-base-nli-max-tokens (mech-f1)	0.110	0.601 ± 0.031	0.497 ± 0.029	0.779 ± 0.022
bert-base-nli-cls-token (ref-prec)	0.212	0.178 ± 0.048	0.148 ± 0.041	0.170 ± 0.058
bert-base-nli-cls-token (ref-f1)	0.194	0.214 ± 0.044	0.167 ± 0.038	0.267 ± 0.054
bert-base-nli-cls-token (sd-prec)	0.183	0.590 ± 0.034	0.496 ± 0.032	0.772 ± 0.024
bert-base-nli-cls-token (sd-f1)	0.109	0.634 ± 0.030	0.524 ± 0.029	0.790 ± 0.021
bert-base-nli-cls-token (mech-prec)	0.174	0.555 ± 0.035	0.471 ± 0.032	0.734 ± 0.027
bert-base-nli-cls-token (mech-f1)	0.111	0.599 ± 0.032	0.495 ± 0.030	0.770 ± 0.022
bert-large-nli-mean-tokens (ref-prec)	0.193	0.169 ± 0.047	0.135 ± 0.040	0.248 ± 0.054
bert-large-nli-mean-tokens (ref-f1)	0.179	0.221 ± 0.044	0.172 ± 0.038	0.342 ± 0.049
bert-large-nli-mean-tokens (sd-prec)	0.182	0.550 ± 0.035	0.458 ± 0.033	0.762 ± 0.025
bert-large-nli-mean-tokens (sd-f1)	0.114	0.596 ± 0.030	0.488 ± 0.029	0.784 ± 0.022
bert-large-nli-mean-tokens (mech-prec)	0.174	0.515 ± 0.037	0.429 ± 0.034	0.728 ± 0.029
bert-large-nli-mean-tokens (mech-f1)	0.116	0.573 ± 0.031	0.468 ± 0.029	0.766 ± 0.023
bert-large-nli-cls-token (ref-prec)	0.191	0.158 ± 0.049	0.121 ± 0.042	0.196 ± 0.057
bert-large-nli-cls-token (ref-f1)	0.177	0.202 ± 0.048	0.160 ± 0.041	0.286 ± 0.054
bert-large-nli-cls-token (sd-prec)	0.195	0.546 ± 0.035	0.451 ± 0.034	0.762 ± 0.024
bert-large-nli-cls-token (sd-f1)	0.124	0.599 ± 0.030	0.493 ± 0.029	0.776 ± 0.022
bert-large-nli-cls-token (mech-prec)	0.187	0.510 ± 0.036	0.421 ± 0.033	0.726 ± 0.028
bert-large-nli-cls-token (mech-f1)	0.126	0.577 ± 0.031	0.473 ± 0.030	0.755 ± 0.024
bert-large-nli-stsb-mean-tokens (ref-prec)	0.320	0.174 ± 0.048	0.145 ± 0.041	0.144 ± 0.057
bert-large-nli-stsb-mean-tokens (ref-f1)	0.303	0.196 ± 0.047	0.156 ± 0.040	0.223 ± 0.054
bert-large-nli-stsb-mean-tokens (sd-prec)	0.132	0.578 ± 0.034	0.487 ± 0.032	0.768 ± 0.025
bert-large-nli-stsb-mean-tokens (sd-f1)	0.180	0.624 ± 0.028	0.515 ± 0.027	0.791 ± 0.022
bert-large-nli-stsb-mean-tokens (mech-prec)	0.137	0.530 ± 0.039	0.451 ± 0.036	0.726 ± 0.030
bert-large-nli-stsb-mean-tokens (mech-f1)	0.180	0.588 ± 0.033	0.488 ± 0.031	0.767 ± 0.023

Table A.12.: Mean Absolute Error mae alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (sem) between human (*Relevance*– Section 3.2) scores and the scores predicted by the supert and the alternative versions, as described in Section 5.2. The correlations are calculated at the document level.

	BioASQ (2018 Val) *Information Precision* estimations				
	Micro	*Macro*			
Method	mae	mae	ρ	τ	r
bert-base-nli-mean-tokens (ref-prec)	**0.199**	**0.197 ± 0.005**	0.340 ± 0.021	0.302 ± 0.019	0.447 ± 0.022
bert-base-nli-mean-tokens (ref-f1)	0.211	0.209 ± 0.005	0.251 ± 0.022	0.225 ± 0.020	0.381 ± 0.024
bert-base-nli-mean-tokens (snip-prec)	0.284	0.281 ± 0.008	0.318 ± 0.020	0.284 ± 0.018	0.449 ± 0.021
bert-base-nli-mean-tokens (snip-f1)	0.253	0.250 ± 0.006	-0.020 ± 0.024	-0.017 ± 0.021	0.245 ± 0.027
bert-base-nli-mean-tokens (sd-prec)	0.220	0.218 ± 0.006	0.256 ± 0.020	0.227 ± 0.018	0.397 ± 0.023
bert-base-nli-mean-tokens (sd-f1)	0.235	0.233 ± 0.004	-0.076 ± 0.024	-0.066 ± 0.022	0.181 ± 0.028
bert-base-nli-mean-tokens (mech-prec)	0.228	0.226 ± 0.006	0.214 ± 0.022	0.190 ± 0.019	0.341 ± 0.024
bert-base-nli-mean-tokens (mech-f1)	0.243	0.240 ± 0.005	0.105 ± 0.023	0.098 ± 0.021	0.271 ± 0.026
bert-base-nli-max-tokens (ref-prec)	0.203	0.201 ± 0.005	0.336 ± 0.021	0.298 ± 0.019	0.442 ± 0.022
bert-base-nli-max-tokens (ref-f1)	0.217	0.214 ± 0.005	0.234 ± 0.022	0.210 ± 0.020	0.373 ± 0.024
bert-base-nli-max-tokens (snip-prec)	0.288	0.285 ± 0.008	0.315 ± 0.020	0.281 ± 0.018	0.438 ± 0.022
bert-base-nli-max-tokens (snip-f1)	0.264	0.261 ± 0.006	-0.032 ± 0.024	-0.027 ± 0.021	0.244 ± 0.027
bert-base-nli-max-tokens (sd-prec)	0.229	0.227 ± 0.006	0.252 ± 0.021	0.224 ± 0.018	0.382 ± 0.023
bert-base-nli-max-tokens (sd-f1)	0.232	0.230 ± 0.005	-0.077 ± 0.024	-0.067 ± 0.022	0.178 ± 0.028
bert-base-nli-max-tokens (mech-prec)	0.235	0.233 ± 0.006	0.206 ± 0.022	0.185 ± 0.019	0.334 ± 0.024
bert-base-nli-max-tokens (mech-f1)	0.250	0.247 ± 0.006	0.094 ± 0.023	0.087 ± 0.021	0.264 ± 0.026
bert-base-nli-cls-token (ref-prec)	0.202	0.199 ± 0.005	0.336 ± 0.021	0.296 ± 0.019	0.450 ± 0.021
bert-base-nli-cls-token (ref-f1)	0.216	0.214 ± 0.005	0.240 ± 0.022	0.213 ± 0.020	0.376 ± 0.024
bert-base-nli-cls-token (snip-prec)	0.291	0.288 ± 0.008	0.320 ± 0.020	0.286 ± 0.018	0.448 ± 0.022
bert-base-nli-max-tokens (snip-f1)	0.262	0.259 ± 0.006	-0.051 ± 0.024	-0.045 ± 0.022	0.233 ± 0.027
bert-base-nli-cls-token (sd-prec)	0.226	0.224 ± 0.006	0.285 ± 0.020	0.252 ± 0.018	0.415 ± 0.022
bert-base-nli-cls-token (sd-f1)	0.232	0.229 ± 0.005	-0.075 ± 0.024	-0.067 ± 0.022	0.183 ± 0.028
bert-base-nli-cls-token (mech-prec)	0.233	0.230 ± 0.006	0.233 ± 0.022	0.208 ± 0.019	0.343 ± 0.023
bert-base-nli-cls-token (mech-f1)	0.246	0.243 ± 0.005	0.113 ± 0.023	0.105 ± 0.021	0.274 ± 0.026
bert-large-nli-mean-tokens (ref-prec)	0.203	0.201 ± 0.005	0.330 ± 0.021	0.295 ± 0.019	0.441 ± 0.022
bert-large-nli-mean-tokens (ref-f1)	0.217	0.215 ± 0.005	0.239 ± 0.022	0.214 ± 0.020	0.376 ± 0.024
bert-large-nli-mean-tokens (snip-prec)	0.287	0.284 ± 0.008	0.319 ± 0.020	0.285 ± 0.018	0.439 ± 0.022
bert-large-nli-mean-tokens (snip-f1)	0.261	0.258 ± 0.006	-0.005 ± 0.023	-0.005 ± 0.021	0.253 ± 0.027
bert-large-nli-mean-tokens (sd-prec)	0.228	0.226 ± 0.006	0.249 ± 0.020	0.220 ± 0.018	0.377 ± 0.023
bert-large-nli-mean-tokens (sd-f1)	0.230	0.228 ± 0.005	-0.077 ± 0.024	-0.067 ± 0.021	0.177 ± 0.028
bert-large-nli-mean-tokens (mech-prec)	0.235	0.232 ± 0.006	0.205 ± 0.022	0.185 ± 0.020	0.327 ± 0.024
bert-large-nli-mean-tokens (mech-f1)	0.248	0.245 ± 0.005	0.113 ± 0.023	0.105 ± 0.021	0.271 ± 0.026
bert-large-nli-cls-token (ref-prec)	0.226	0.223 ± 0.005	0.326 ± 0.021	0.289 ± 0.019	0.436 ± 0.022
bert-large-nli-cls-token (ref-f1)	0.239	0.237 ± 0.006	0.241 ± 0.022	0.215 ± 0.020	0.373 ± 0.024
bert-large-nli-cls-token (snip-prec)	0.296	0.293 ± 0.008	0.319 ± 0.020	0.285 ± 0.018	0.437 ± 0.022
bert-large-nli-cls-token (snip-f1)	0.278	0.275 ± 0.007	-0.004 ± 0.023	-0.003 ± 0.021	0.255 ± 0.027
bert-large-nli-cls-token (sd-prec)	0.250	0.248 ± 0.006	0.278 ± 0.020	0.246 ± 0.018	0.397 ± 0.023
bert-large-nli-cls-token (sd-f1)	0.242	0.239 ± 0.005	-0.040 ± 0.024	-0.033 ± 0.021	0.205 ± 0.027
bert-large-nli-cls-token (mech-prec)	0.252	0.250 ± 0.006	0.230 ± 0.022	0.207 ± 0.019	0.334 ± 0.024
bert-large-nli-cls-token (mech-f1)	0.266	0.263 ± 0.006	0.144 ± 0.023	0.131 ± 0.020	0.297 ± 0.025
bert-large-nli-stsb-mean-tokens (ref-prec)	0.201	0.200 ± 0.004	**0.355 ± 0.021**	**0.315 ± 0.019**	**0.463 ± 0.021**
bert-large-nli-stsb-mean-tokens (ref-f1)	0.211	0.209 ± 0.004	0.279 ± 0.021	0.249 ± 0.019	0.402 ± 0.023
bert-large-nli-stsb-mean-tokens (snip-prec)	0.271	0.268 ± 0.007	0.330 ± 0.020	0.295 ± 0.018	0.451 ± 0.022
bert-large-nli-stsb-mean-tokens (snip-f1)	0.236	0.233 ± 0.005	0.007 ± 0.023	0.008 ± 0.021	0.254 ± 0.027
bert-large-nli-stsb-mean-tokens (sd-prec)	0.215	0.212 ± 0.005	0.276 ± 0.020	0.247 ± 0.018	0.405 ± 0.022
bert-large-nli-stsb-mean-tokens (sd-f1)	0.247	0.245 ± 0.004	-0.047 ± 0.024	-0.040 ± 0.021	0.190 ± 0.027
bert-large-nli-stsb-mean-tokens (mech-prec)	0.226	0.225 ± 0.005	0.217 ± 0.022	0.194 ± 0.019	0.327 ± 0.023
bert-large-nli-stsb-mean-tokens (mech-f1)	0.239	0.237 ± 0.005	0.120 ± 0.023	0.112 ± 0.021	0.270 ± 0.026

Table A.13.: Mean Absolute Error mae on the *micro* level and *macro* mae alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (sem) between human (*Information Precision*– Section 3.3) scores and the scores predicted by the supert and the alternative versions, as described in Section 5.2. The correlations are calculated at the document level.

	BioASQ (2019 Test) *Information Precision* estimations				
	$Micro$	$Macro$			
Method	MAE	MAE	ρ	τ	r
bert-base-nli-mean-tokens (ref-prec)	**0.187**	**0.184** ± **0.005**	0.326 ± 0.030	0.299 ± 0.028	0.387 ± 0.031
bert-base-nli-mean-tokens (ref-f1)	0.190	0.188 ± 0.005	0.295 ± 0.030	0.273 ± 0.029	0.351 ± 0.032
bert-base-nli-mean-tokens (snip-prec)	0.190	0.184 ± 0.007	0.195 ± 0.032	0.183 ± 0.030	0.297 ± 0.034
bert-base-nli-mean-tokens (snip-f1)	0.186	0.181 ± 0.005	0.011 ± 0.035	0.004 ± 0.033	0.149 ± 0.039
bert-base-nli-mean-tokens (sd-prec)	0.181	0.172 ± 0.005	0.149 ± 0.033	0.139 ± 0.031	0.260 ± 0.034
bert-base-nli-mean-tokens (sd-f1)	0.228	0.222 ± 0.005	-0.044 ± 0.035	-0.048 ± 0.033	0.099 ± 0.038
bert-base-nli-mean-tokens (mech-prec)	0.198	0.191 ± 0.006	0.217 ± 0.032	0.207 ± 0.030	0.297 ± 0.033
bert-base-nli-mean-tokens (mech-f1)	0.216	0.210 ± 0.005	0.021 ± 0.035	0.017 ± 0.033	0.139 ± 0.037
bert-base-nli-max-tokens (ref-prec)	0.180	0.177 ± 0.006	0.329 ± 0.030	0.303 ± 0.028	0.401 ± 0.031
bert-base-nli-max-tokens (ref-f1)	0.185	0.182 ± 0.005	0.286 ± 0.031	0.264 ± 0.029	0.342 ± 0.033
bert-base-nli-max-tokens (snip-prec)	0.191	0.184 ± 0.007	0.181 ± 0.032	0.170 ± 0.030	0.287 ± 0.034
bert-base-nli-max-tokens (snip-f1)	0.188	0.183 ± 0.006	-0.007 ± 0.035	-0.012 ± 0.033	0.148 ± 0.039
bert-base-nli-max-tokens (sd-prec)	0.180	0.172 ± 0.005	0.142 ± 0.032	0.132 ± 0.030	0.254 ± 0.034
bert-base-nli-max-tokens (sd-f1)	0.212	0.205 ± 0.005	-0.052 ± 0.035	-0.055 ± 0.033	0.105 ± 0.038
bert-base-nli-max-tokens (mech-prec)	0.199	0.191 ± 0.007	0.219 ± 0.033	0.209 ± 0.031	0.301 ± 0.033
bert-base-nli-max-tokens (mech-f1)	0.211	0.205 ± 0.005	0.020 ± 0.034	0.016 ± 0.032	0.134 ± 0.037
bert-base-nli-cls-token (ref-prec)	0.177	0.172 ± 0.005	0.330 ± 0.030	0.303 ± 0.028	0.410 ± 0.031
bert-base-nli-cls-token (ref-f1)	0.180	0.177 ± 0.005	0.299 ± 0.031	0.278 ± 0.029	0.360 ± 0.033
bert-base-nli-cls-token (snip-prec)	0.193	0.185 ± 0.007	0.189 ± 0.032	0.177 ± 0.030	0.294 ± 0.034
bert-base-nli-max-tokens (snip-f1)	0.187	0.182 ± 0.005	-0.023 ± 0.035	-0.028 ± 0.033	0.135 ± 0.039
bert-base-nli-cls-token (sd-prec)	0.176	0.168 ± 0.005	0.192 ± 0.032	0.178 ± 0.030	0.298 ± 0.033
bert-base-nli-cls-token (sd-f1)	0.211	0.204 ± 0.005	-0.048 ± 0.035	-0.052 ± 0.033	0.099 ± 0.038
bert-base-nli-cls-token (mech-prec)	0.196	0.189 ± 0.006	0.230 ± 0.032	0.218 ± 0.030	0.302 ± 0.032
bert-base-nli-cls-token (mech-f1)	0.209	0.202 ± 0.005	0.045 ± 0.035	0.040 ± 0.033	0.146 ± 0.037
bert-large-nli-mean-tokens (ref-prec)	0.176	0.172 ± 0.005	0.335 ± 0.030	0.309 ± 0.028	0.395 ± 0.032
bert-large-nli-mean-tokens (ref-f1)	0.180	0.177 ± 0.005	0.287 ± 0.031	0.265 ± 0.030	0.358 ± 0.032
bert-large-nli-mean-tokens (snip-prec)	0.191	0.184 ± 0.007	0.176 ± 0.033	0.164 ± 0.031	0.281 ± 0.035
bert-large-nli-mean-tokens (snip-f1)	0.188	0.182 ± 0.006	0.009 ± 0.035	0.003 ± 0.033	0.152 ± 0.038
bert-large-nli-mean-tokens (sd-prec)	0.179	0.171 ± 0.005	0.164 ± 0.033	0.154 ± 0.031	0.261 ± 0.034
bert-large-nli-mean-tokens (sd-f1)	0.179	0.170 ± 0.006	0.155 ± 0.034	0.148 ± 0.032	0.247 ± 0.035
bert-large-nli-mean-tokens (mech-prec)	0.198	0.190 ± 0.006	0.211 ± 0.033	0.203 ± 0.031	0.296 ± 0.033
bert-large-nli-mean-tokens (mech-f1)	0.208	0.201 ± 0.005	0.051 ± 0.035	0.046 ± 0.033	0.157 ± 0.037
bert-large-nli-cls-token (ref-prec)	0.168	0.162 ± 0.005	0.328 ± 0.030	0.301 ± 0.029	0.396 ± 0.032
bert-large-nli-cls-token (ref-f1)	0.174	0.169 ± 0.005	0.301 ± 0.031	0.279 ± 0.030	0.371 ± 0.033
bert-large-nli-cls-token (snip-prec)	0.194	0.187 ± 0.007	0.192 ± 0.032	0.178 ± 0.031	0.291 ± 0.035
bert-large-nli-cls-token (snip-f1)	0.190	0.184 ± 0.006	0.011 ± 0.035	0.004 ± 0.033	0.155 ± 0.038
bert-large-nli-cls-token (sd-prec)	0.184	0.175 ± 0.006	0.203 ± 0.033	0.189 ± 0.031	0.290 ± 0.034
bert-large-nli-cls-token (sd-f1)	0.195	0.187 ± 0.005	-0.027 ± 0.035	-0.032 ± 0.033	0.109 ± 0.038
bert-large-nli-cls-token (mech-prec)	0.199	0.190 ± 0.007	0.228 ± 0.032	0.219 ± 0.030	0.315 ± 0.032
bert-large-nli-cls-token (mech-f1)	0.205	0.197 ± 0.005	0.073 ± 0.035	0.068 ± 0.033	0.167 ± 0.037
bert-large-nli-stsb-mean-tokens (ref-prec)	0.213	0.211 ± 0.006	**0.361** ± **0.029**	**0.332** ± **0.028**	**0.421** ± **0.031**
bert-large-nli-stsb-mean-tokens (ref-f1)	0.215	0.214 ± 0.006	0.301 ± 0.032	0.279 ± 0.030	0.378 ± 0.032
bert-large-nli-stsb-mean-tokens (snip-prec)	0.188	0.181 ± 0.007	0.191 ± 0.032	0.178 ± 0.031	0.287 ± 0.034
bert-large-nli-stsb-mean-tokens (snip-f1)	0.188	0.183 ± 0.005	0.015 ± 0.036	0.009 ± 0.034	0.153 ± 0.038
bert-large-nli-stsb-mean-tokens (sd-prec)	0.196	0.189 ± 0.005	0.184 ± 0.033	0.170 ± 0.031	0.270 ± 0.033
bert-large-nli-stsb-mean-tokens (sd-f1)	0.278	0.273 ± 0.005	-0.047 ± 0.035	-0.050 ± 0.033	0.095 ± 0.038
bert-large-nli-stsb-mean-tokens (mech-prec)	0.196	0.188 ± 0.006	0.216 ± 0.032	0.206 ± 0.031	0.281 ± 0.033
bert-large-nli-stsb-mean-tokens (mech-f1)	0.245	0.239 ± 0.005	0.061 ± 0.035	0.054 ± 0.033	0.143 ± 0.037

Table A.14.: Mean Absolute Error MAE on the $micro$ level and $macro$ MAE alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (SEM) between human (*Information Precision*– Section 3.3) scores and the scores predicted by the SUPERT and the alternative versions, as described in Section 5.2. The correlations are calculated at the document level. **BOLD** indicates the performance of the corresponding model that best performed on the validation data with respect the correlation level or the MAE.

	BioASQ (2018 Val) *Information Recall* estimations				
	$Micro$	$Macro$			
Method	mae	mae	ρ	τ	r
bert-base-nli-mean-tokens (ref-rec)	0.189	0.188 ± 0.004	0.561 ± 0.015	0.502 ± 0.014	0.669 ± 0.015
bert-base-nli-mean-tokens (ref-f1)	0.212	0.212 ± 0.004	0.483 ± 0.017	0.426 ± 0.016	0.619 ± 0.017
bert-base-nli-mean-tokens (snip-rec)	0.200	0.198 ± 0.005	0.570 ± 0.015	0.508 ± 0.014	0.666 ± 0.016
bert-base-nli-mean-tokens (snip-f1)	0.188	0.186 ± 0.005	0.547 ± 0.015	0.486 ± 0.013	0.657 ± 0.016
bert-base-nli-mean-tokens (sd-rec)	0.293	0.293 ± 0.005	0.528 ± 0.016	0.471 ± 0.014	0.613 ± 0.018
bert-base-nli-mean-tokens (sd-f1)	0.260	0.259 ± 0.004	0.531 ± 0.016	0.471 ± 0.014	0.639 ± 0.017
bert-base-nli-mean-tokens (mech-rec)	0.261	0.261 ± 0.004	0.498 ± 0.017	0.443 ± 0.015	0.595 ± 0.019
bert-base-nli-mean-tokens (mech-f1)	0.234	0.233 ± 0.005	0.381 ± 0.020	0.338 ± 0.018	0.558 ± 0.020
bert-base-nli-max-tokens (ref-rec)	0.181	0.181 ± 0.004	0.568 ± 0.015	0.510 ± 0.014	0.668 ± 0.016
bert-base-nli-max-tokens (ref-f1)	0.201	0.201 ± 0.004	0.485 ± 0.017	0.428 ± 0.016	0.622 ± 0.018
bert-base-nli-max-tokens (snip-rec)	0.190	0.189 ± 0.005	0.572 ± 0.015	0.510 ± 0.014	0.661 ± 0.016
bert-base-nli-max-tokens (snip-f1)	0.186	0.184 ± 0.005	0.548 ± 0.015	0.487 ± 0.014	0.651 ± 0.017
bert-base-nli-max-tokens (sd-rec)	0.268	0.267 ± 0.004	0.525 ± 0.016	0.467 ± 0.015	0.609 ± 0.018
bert-base-nli-max-tokens (sd-f1)	0.239	0.238 ± 0.004	0.528 ± 0.016	0.468 ± 0.015	0.628 ± 0.018
bert-base-nli-max-tokens (mech-rec)	0.242	0.242 ± 0.004	0.497 ± 0.017	0.443 ± 0.016	0.594 ± 0.019
bert-base-nli-max-tokens (mech-f1)	0.222	0.222 ± 0.005	0.373 ± 0.020	0.331 ± 0.018	0.551 ± 0.021
bert-base-nli-cls-token (ref-rec)	**0.179**	0.179 ± 0.004	**0.592 ± 0.014**	**0.531 ± 0.013**	**0.693 ± 0.015**
bert-base-nli-cls-token (ref-f1)	0.202	0.202 ± 0.004	0.504 ± 0.017	0.444 ± 0.015	0.638 ± 0.017
bert-base-nli-cls-token (snip-rec)	0.195	0.193 ± 0.005	0.575 ± 0.015	0.512 ± 0.014	0.672 ± 0.016
bert-base-nli-max-tokens (snip-f1)	0.188	0.186 ± 0.005	0.560 ± 0.015	0.499 ± 0.013	0.669 ± 0.016
bert-base-nli-cls-token (sd-rec)	0.274	0.274 ± 0.004	0.534 ± 0.016	0.476 ± 0.015	0.622 ± 0.017
bert-base-nli-cls-token (sd-f1)	0.243	0.243 ± 0.004	0.550 ± 0.016	0.488 ± 0.014	0.652 ± 0.016
bert-base-nli-cls-token (mech-rec)	0.248	0.248 ± 0.004	0.506 ± 0.017	0.451 ± 0.015	0.602 ± 0.018
bert-base-nli-cls-token (mech-f1)	0.225	0.225 ± 0.004	0.374 ± 0.021	0.331 ± 0.019	0.559 ± 0.020
bert-large-nli-mean-tokens (ref-rec)	0.185	0.184 ± 0.004	0.560 ± 0.015	0.501 ± 0.014	0.659 ± 0.016
bert-large-nli-mean-tokens (ref-f1)	0.203	0.203 ± 0.004	0.465 ± 0.017	0.409 ± 0.016	0.615 ± 0.017
bert-large-nli-mean-tokens (snip-rec)	0.194	0.192 ± 0.005	0.561 ± 0.015	0.500 ± 0.014	0.659 ± 0.016
bert-large-nli-mean-tokens (snip-f1)	0.188	0.186 ± 0.005	0.522 ± 0.015	0.465 ± 0.014	0.644 ± 0.017
bert-large-nli-mean-tokens (sd-rec)	0.267	0.266 ± 0.004	0.525 ± 0.016	0.467 ± 0.014	0.608 ± 0.018
bert-large-nli-mean-tokens (sd-f1)	0.240	0.239 ± 0.004	0.531 ± 0.015	0.470 ± 0.014	0.633 ± 0.017
bert-large-nli-mean-tokens (mech-rec)	0.242	0.242 ± 0.004	0.495 ± 0.017	0.440 ± 0.015	0.590 ± 0.019
bert-large-nli-mean-tokens (mech-f1)	0.223	0.222 ± 0.005	0.370 ± 0.020	0.329 ± 0.018	0.548 ± 0.020
bert-large-nli-cls-token (ref-rec)	0.179	**0.178 ± 0.005**	0.578 ± 0.015	0.520 ± 0.013	0.662 ± 0.016
bert-large-nli-cls-token (ref-f1)	0.192	0.191 ± 0.005	0.474 ± 0.018	0.418 ± 0.016	0.618 ± 0.018
bert-large-nli-cls-token (snip-rec)	0.190	0.188 ± 0.005	0.568 ± 0.015	0.507 ± 0.014	0.663 ± 0.016
bert-large-nli-cls-token (snip-f1)	0.191	0.188 ± 0.006	0.537 ± 0.015	0.478 ± 0.014	0.649 ± 0.016
bert-large-nli-cls-token (sd-rec)	0.235	0.234 ± 0.004	0.529 ± 0.016	0.470 ± 0.015	0.611 ± 0.018
bert-large-nli-cls-token (sd-f1)	0.220	0.219 ± 0.005	0.531 ± 0.016	0.469 ± 0.014	0.632 ± 0.017
bert-large-nli-cls-token (mech-rec)	0.223	0.222 ± 0.004	0.492 ± 0.017	0.438 ± 0.015	0.591 ± 0.018
bert-large-nli-cls-token (mech-f1)	0.216	0.214 ± 0.005	0.365 ± 0.021	0.324 ± 0.018	0.542 ± 0.021
bert-large-nli-stsb-mean-tokens (ref-rec)	0.211	0.212 ± 0.005	0.573 ± 0.014	0.514 ± 0.013	0.670 ± 0.015
bert-large-nli-stsb-mean-tokens (ref-f1)	0.240	0.241 ± 0.005	0.475 ± 0.018	0.419 ± 0.016	0.622 ± 0.017
bert-large-nli-stsb-mean-tokens (snip-rec)	0.217	0.216 ± 0.005	0.569 ± 0.015	0.507 ± 0.013	0.672 ± 0.015
bert-large-nli-stsb-mean-tokens (snip-f1)	0.193	0.192 ± 0.005	0.538 ± 0.015	0.478 ± 0.013	0.661 ± 0.016
bert-large-nli-stsb-mean-tokens (sd-rec)	0.351	0.351 ± 0.006	0.535 ± 0.016	0.475 ± 0.014	0.624 ± 0.017
bert-large-nli-stsb-mean-tokens (sd-f1)	0.307	0.307 ± 0.005	0.536 ± 0.015	0.474 ± 0.014	0.648 ± 0.016
bert-large-nli-stsb-mean-tokens (mech-rec)	0.303	0.304 ± 0.005	0.491 ± 0.017	0.436 ± 0.016	0.592 ± 0.018
bert-large-nli-stsb-mean-tokens (mech-f1)	0.262	0.263 ± 0.005	0.380 ± 0.020	0.337 ± 0.018	0.557 ± 0.020

Table A.15.: Mean Absolute Error mae on the $micro$ level and $macro$ mae alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (sem) between human (*Information Recall*– Section 3.3) scores and the scores predicted by the supert and the alternative versions, as described in Section 5.2. The correlations are calculated at the document level.

	BioASQ (2019 Test) *Information Recall* estimations				
	Micro	*Macro*			
Method	MAE	MAE	ρ	τ	r
BERT-BASE-NLI-MEAN-TOKENS (REF-REC)	0.196	0.196 ± 0.006	0.638 ± 0.020	0.596 ± 0.019	0.710 ± 0.022
BERT-BASE-NLI-MEAN-TOKENS (REF-F1)	0.216	0.217 ± 0.006	0.526 ± 0.024	0.483 ± 0.023	0.609 ± 0.026
BERT-BASE-NLI-MEAN-TOKENS (SNIP-REC)	0.179	0.181 ± 0.005	0.650 ± 0.019	0.604 ± 0.018	0.731 ± 0.019
BERT-BASE-NLI-MEAN-TOKENS (SNIP-F1)	0.164	0.166 ± 0.006	0.637 ± 0.019	0.590 ± 0.019	0.721 ± 0.020
BERT-BASE-NLI-MEAN-TOKENS (SD-REC)	0.314	0.309 ± 0.005	0.602 ± 0.021	0.557 ± 0.020	0.700 ± 0.020
BERT-BASE-NLI-MEAN-TOKENS (SD-F1)	0.261	0.259 ± 0.005	0.593 ± 0.021	0.548 ± 0.020	0.681 ± 0.021
BERT-BASE-NLI-MEAN-TOKENS (MECH-REC)	0.281	0.276 ± 0.005	0.602 ± 0.021	0.557 ± 0.020	0.685 ± 0.021
BERT-BASE-NLI-MEAN-TOKENS (MECH-F1)	0.232	0.230 ± 0.005	0.492 ± 0.026	0.455 ± 0.025	0.614 ± 0.025
BERT-BASE-NLI-MAX-TOKENS (REF-REC)	0.186	0.186 ± 0.006	0.642 ± 0.020	0.598 ± 0.019	0.722 ± 0.020
BERT-BASE-NLI-MAX-TOKENS (REF-F1)	0.205	0.206 ± 0.006	0.523 ± 0.024	0.479 ± 0.023	0.622 ± 0.025
BERT-BASE-NLI-MAX-TOKENS (SNIP-REC)	0.169	0.171 ± 0.005	0.655 ± 0.018	0.608 ± 0.018	0.736 ± 0.019
BERT-BASE-NLI-MAX-TOKENS (SNIP-F1)	0.160	0.162 ± 0.006	0.640 ± 0.019	0.594 ± 0.018	0.721 ± 0.020
BERT-BASE-NLI-MAX-TOKENS (SD-REC)	0.285	0.280 ± 0.005	0.609 ± 0.021	0.565 ± 0.020	0.701 ± 0.020
BERT-BASE-NLI-MAX-TOKENS (SD-F1)	0.239	0.237 ± 0.005	0.606 ± 0.021	0.560 ± 0.020	0.689 ± 0.020
BERT-BASE-NLI-MAX-TOKENS (MECH-REC)	0.257	0.252 ± 0.005	0.605 ± 0.021	0.561 ± 0.020	0.684 ± 0.021
BERT-BASE-NLI-MAX-TOKENS (MECH-F1)	0.217	0.215 ± 0.005	0.498 ± 0.026	0.460 ± 0.024	0.618 ± 0.025
BERT-BASE-NLI-CLS-TOKEN (REF-REC)	**0.177**	0.178 ± 0.005	**0.655 ± 0.019**	**0.611 ± 0.019**	**0.746 ± 0.019**
BERT-BASE-NLI-CLS-TOKEN (REF-F1)	0.202	0.203 ± 0.005	0.568 ± 0.022	0.523 ± 0.021	0.662 ± 0.023
BERT-BASE-NLI-CLS-TOKEN (SNIP-REC)	0.173	0.174 ± 0.005	0.655 ± 0.019	0.608 ± 0.018	0.744 ± 0.018
BERT-BASE-NLI-MAX-TOKENS (SNIP-F1)	0.162	0.164 ± 0.006	0.639 ± 0.020	0.592 ± 0.019	0.732 ± 0.020
BERT-BASE-NLI-CLS-TOKEN (SD-REC)	0.291	0.286 ± 0.005	0.620 ± 0.020	0.575 ± 0.019	0.712 ± 0.020
BERT-BASE-NLI-CLS-TOKEN (SD-F1)	0.243	0.241 ± 0.005	0.615 ± 0.020	0.569 ± 0.019	0.703 ± 0.020
BERT-BASE-NLI-CLS-TOKEN (MECH-REC)	0.264	0.259 ± 0.005	0.611 ± 0.021	0.567 ± 0.020	0.691 ± 0.021
BERT-BASE-NLI-CLS-TOKEN (MECH-F1)	0.221	0.218 ± 0.005	0.484 ± 0.026	0.447 ± 0.025	0.607 ± 0.025
BERT-LARGE-NLI-MEAN-TOKENS (REF-REC)	0.185	0.185 ± 0.006	0.628 ± 0.021	0.587 ± 0.020	0.702 ± 0.021
BERT-LARGE-NLI-MEAN-TOKENS (REF-F1)	0.203	0.203 ± 0.006	0.518 ± 0.025	0.477 ± 0.024	0.600 ± 0.026
BERT-LARGE-NLI-MEAN-TOKENS (SNIP-REC)	0.171	0.172 ± 0.005	0.656 ± 0.019	0.609 ± 0.018	0.732 ± 0.019
BERT-LARGE-NLI-MEAN-TOKENS (SNIP-F1)	0.162	0.164 ± 0.006	0.640 ± 0.019	0.593 ± 0.018	0.727 ± 0.019
BERT-LARGE-NLI-MEAN-TOKENS (SD-REC)	0.279	0.275 ± 0.005	0.617 ± 0.020	0.571 ± 0.020	0.700 ± 0.021
BERT-LARGE-NLI-MEAN-TOKENS (SD-F1)	0.236	0.235 ± 0.005	0.600 ± 0.021	0.555 ± 0.020	0.683 ± 0.021
BERT-LARGE-NLI-MEAN-TOKENS (MECH-REC)	0.254	0.250 ± 0.005	0.608 ± 0.021	0.564 ± 0.020	0.691 ± 0.021
BERT-LARGE-NLI-MEAN-TOKENS (MECH-F1)	0.216	0.214 ± 0.005	0.485 ± 0.026	0.452 ± 0.025	0.603 ± 0.026
BERT-LARGE-NLI-CLS-TOKEN (REF-REC)	0.167	**0.168 ± 0.006**	0.655 ± 0.019	0.613 ± 0.018	0.740 ± 0.019
BERT-LARGE-NLI-CLS-TOKEN (REF-F1)	0.183	0.184 ± 0.006	0.548 ± 0.023	0.504 ± 0.022	0.646 ± 0.024
BERT-LARGE-NLI-CLS-TOKEN (SNIP-REC)	0.162	0.164 ± 0.006	0.650 ± 0.019	0.603 ± 0.018	0.733 ± 0.019
BERT-LARGE-NLI-CLS-TOKEN (SNIP-F1)	0.162	0.164 ± 0.006	0.639 ± 0.019	0.593 ± 0.018	0.723 ± 0.020
BERT-LARGE-NLI-CLS-TOKEN (SD-REC)	0.237	0.235 ± 0.005	0.612 ± 0.021	0.567 ± 0.020	0.692 ± 0.021
BERT-LARGE-NLI-CLS-TOKEN (SD-F1)	0.210	0.210 ± 0.005	0.613 ± 0.020	0.566 ± 0.019	0.682 ± 0.020
BERT-LARGE-NLI-CLS-TOKEN (MECH-REC)	0.222	0.220 ± 0.005	0.603 ± 0.021	0.559 ± 0.020	0.680 ± 0.021
BERT-LARGE-NLI-CLS-TOKEN (MECH-F1)	0.201	0.199 ± 0.005	0.483 ± 0.027	0.449 ± 0.025	0.589 ± 0.026
BERT-LARGE-NLI-STSB-MEAN-TOKENS (REF-REC)	0.223	0.220 ± 0.006	0.642 ± 0.019	0.600 ± 0.019	0.736 ± 0.019
BERT-LARGE-NLI-STSB-MEAN-TOKENS (REF-F1)	0.248	0.246 ± 0.006	0.547 ± 0.024	0.506 ± 0.022	0.649 ± 0.023
BERT-LARGE-NLI-STSB-MEAN-TOKENS (SNIP-REC)	0.201	0.202 ± 0.006	0.654 ± 0.019	0.607 ± 0.018	0.740 ± 0.019
BERT-LARGE-NLI-STSB-MEAN-TOKENS (SNIP-F1)	0.173	0.174 ± 0.005	0.641 ± 0.019	0.595 ± 0.019	0.733 ± 0.020
BERT-LARGE-NLI-STSB-MEAN-TOKENS (SD-REC)	0.383	0.377 ± 0.007	0.622 ± 0.020	0.577 ± 0.019	0.702 ± 0.020
BERT-LARGE-NLI-STSB-MEAN-TOKENS (SD-F1)	0.318	0.314 ± 0.006	0.609 ± 0.021	0.564 ± 0.020	0.684 ± 0.021
BERT-LARGE-NLI-STSB-MEAN-TOKENS (MECH-REC)	0.337	0.330 ± 0.006	0.612 ± 0.021	0.569 ± 0.020	0.686 ± 0.021
BERT-LARGE-NLI-STSB-MEAN-TOKENS (MECH-F1)	0.272	0.268 ± 0.005	0.506 ± 0.025	0.469 ± 0.024	0.607 ± 0.025

Table A.16.: Mean Absolute Error MAE on the *micro* level and *macro* MAE alongside the Spearman's ρ, Kendall's τ and Pearson's r correlations ± Standard Error of the Mean (SEM) between human (*Information Recall*– Section 3.3) scores and the scores predicted by the SUPERT and the alternative versions, as described in Section 5.2. The correlations are calculated at the document level. **BOLD** indicates the performance of the corresponding model that best performed on the validation data with respect the correlation level or the MAE.