A Closer Look at Claim Decomposition

Miriam Wanner*, **Seth Ebner***, **Zhengping Jiang**,
Mark Dredze, **Benjamin Van Durme**

Johns Hopkins University

{mwanner5,seth,zjiang31,mdredze,vandurme}@jhu.edu

Abstract

As generated text becomes more commonplace, it is increasingly important to evaluate how well-supported such text is by external knowledge sources. Many approaches for evaluating textual support rely on some method for decomposing text into its individual subclaims which are scored against a trusted reference. We investigate how various methods of claim decomposition—especially LLM-based methods—affect the result of an evaluation approach such as the recently proposed FACTSCORE, finding that it is sensitive to the decomposition method used. This sensitivity arises because such metrics attribute overall textual support to the model that generated the text even though error can also come from the metric's decomposition step. To measure decomposition quality, we introduce an adaptation of FACTSCORE, which we call DECOMPSCORE. We then propose an LLM-based approach to generating decompositions inspired by Bertrand Russell's theory of logical atomism and neo-Davidsonian semantics and demonstrate its improved decomposition quality over previous methods.

1 Introduction

Recent work uses claim decomposition to determine how well supported a claim is for applications in factual precision of generated text (Min et al., 2023), entailment of human generated text (Kamoi et al., 2023; Chen et al., 2023b), and claim verification (Chen et al., 2023a; Li et al., 2023; Milbauer et al., 2023; Tang et al., 2024), with similar ideas going back over a decade (Hickl and Bensley, 2007). In each of these cases, a claim is decomposed into natural language subclaims,[1] typically using a large language model (LLM), and each subclaim is then scored or aligned to information from external sources using a task-specific metric.

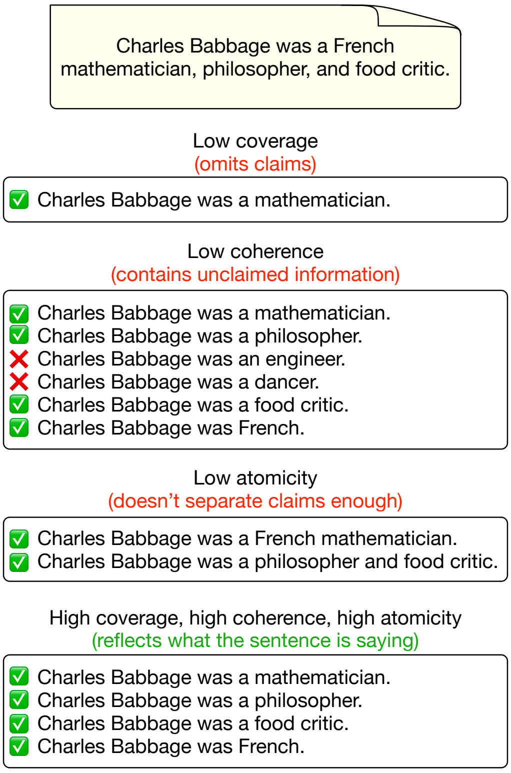
Charles Babbage was a French mathematician, philosopher, and food critic.
Low coverage
(omits claims)
✓ Charles Babbage was a mathematician.
Low coherence
(contains unclaimed information)
✓ Charles Babbage was a mathematician.
✓ Charles Babbage was a philosopher.
✗ Charles Babbage was an engineer.
✗ Charles Babbage was a dancer.
✓ Charles Babbage was a food critic.
✓ Charles Babbage was French.
Low atomicity
(doesn't separate claims enough)
✓ Charles Babbage was a French mathematician.
✓ Charles Babbage was a philosopher and food critic.
High coverage, high coherence, high atomicity
(reflects what the sentence is saying)
✓ Charles Babbage was a mathematician.
✓ Charles Babbage was a philosopher.
✓ Charles Babbage was a food critic.
✓ Charles Babbage was French.

Figure 1: Modes of claim decomposition. The extent to which textual support can be determined depends on how the generated text (yellow box) is decomposed into its subclaims (white boxes). Higher quality decompositions enable more complete identification of discrepancies between generated text and a reference (not shown), which consequently increases the reliability of the downstream textual support metric. Checks and Xs denote that the statement is claimed or is not claimed, respectively, by the generated text.

Claim decompositions with various characteristics are shown in Figure 1. Coverage denotes how much of the information in the claim is present in the subclaims, coherence denotes whether the information in the subclaims accurately reflects what is stated in the claim, and atomicity denotes how

*Equal contribution

[1] The terms "atomic fact" and "atomic proposition" are also used for similar concepts.

*Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*, pages 153 - 175
June 20-21, 2024 ©2024 Association for Computational Linguistics

separated the information in each subclaim is.

Evaluating subclaims individually, as opposed to the entire claim at once, we can assign partial credit to a claim (e.g., for partial support), identify which parts of the claim differ from reference texts (such as a retrieved or pre-specified document or passage), and more easily identify relevant source material for each part of the claim.[2] Claims can come from human-authored text based on cited documents (Kamoi et al., 2023; Chen et al., 2023b,c) or from machine-generated text based on dynamically provided grounding text or text observed during pre-training (Min et al., 2023).

Since claim decomposition determines the number and scope of each evaluated subclaim, any analysis or resulting metric will be inherently tied to the decomposition method. Nevertheless, prior work has left decomposition itself largely untested. How do different decomposition strategies affect downstream analysis? What are their qualitative and quantitative similarities and differences?

We show that a downstream metric of textual support such as FActScore (Min et al., 2023) is sensitive to the decomposition method it uses (Figure 2). While FActScore aims to measure the factual precision of generated text, the number and nature of the subclaims it evaluates from that text depend on the metric's claim decomposition method. The higher the quality of the decomposition method, and the better we understand its characteristics, the more we can attribute the factual precision that FActScore aims to measure to the text generation model rather than to artifacts of the decomposition.

Finding that the method of claim decomposition matters, we introduce DecompScore, an adaptation of FActScore that measures decomposition quality, an important step in determining the reliability of the downstream metric. DecompScore measures the number of subclaims supported by the original claim that was decomposed. Because a decomposition with high atomicity and coverage will have more subclaims than a decomposition that doesn't, we then favor the decomposition method with the greatest DecompScore, especially when coupled with qualitative evidence of high atomicity and coverage.

With a way to compare decomposition methods in hand, we propose an LLM-based decomposition approach inspired by Bertrand Russell's theory of logical atomism and neo-Davidsonian semantics. Our approach gives far more subclaims than other methods while maintaining high coherence with the claim being decomposed, and thus results in greater confidence in the entire pipeline for evaluating the level of textual support.

Our contributions are:

1. Empirical evidence that the method of claim decomposition affects a downstream metric of textual support;
2. Quantitative and qualitative comparisons of claim decomposition methods;
3. A method for claim decomposition inspired by philosophical and semantic theories that outperforms previous methods.

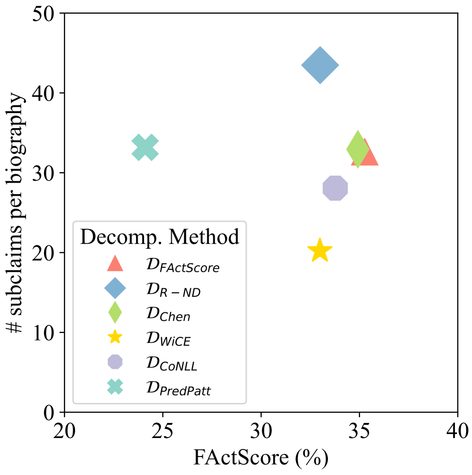

Figure 2: FActScore (macro-averaged across LM$_{\text{SUBJ}}$) using different decomposition methods. The same underlying set of documents is assigned different FActScore values depending on the decomposition method used.

2 Localized Textual Support

FActScore (Min et al., 2023) and WiCE (Kamoi et al., 2023) are representative examples of current LLM-based approaches for determining support for particular claims for different downstream use cases. Broadly, methods of this type decompose a claim into its subclaims, evaluate each subclaim for its level of support based on external sources,

[2] For example, separating the claim "Charles Babbage was a French mathematician" into the atomic subclaims "Charles Babbage was French" and "Charles Babbage was a mathematician" enables a claim verification system to determine that the subclaim about his occupation is supported by trusted reference documents and that the subclaim about his nationality is not supported. The non-atomic original claim as written, however, is not supported.

Name	Instruction	In-Context Examples			
		Static	Dynamic	Sentences	Decompositions
$\mathcal{D}_{\text{FACTSCORE}}$	"Please breakdown the following sentence into independent facts:" (Min et al., 2023)	7	1	Min et al. (2023)	Min et al. (2023)
$\mathcal{D}_{\text{WICE}}$	"Segment the following sentence into individual facts:" (Kamoi et al., 2023)	6	0	Kamoi et al. (2023)	Kamoi et al. (2023)
$\mathcal{D}_{\text{Chen et al.}}$	"Given the following sentence, tell me what claims they are making. Please split the sentence as much as possible, but do not include information not in the sentence:" (Chen et al., 2023c)	7	1	Min et al. (2023)	Min et al. (2023)
$\mathcal{D}_{\text{CoNLL-U}}$	"The sentence below is given in CoNLL-U format. Word lines contain the annotation of a word/token/node in 10 fields separated by single tab characters. Sentences consist of one or more word lines. Please break down the following sentence given in CoNLL-U format into independent facts:"	1	1	Min et al. (2023) + CoNLL-U Parse	Min et al. (2023)
$\mathcal{D}_{\text{R-ND}}$	"Please decompose the following sentence into individual facts:"	7	1	Min et al. (2023)	**Manual (ours)**

Table 1: Summary of LLM prompted claim decomposition methods used in this work (method names are prefixed with $\mathcal{D}$ for "decomposer"). The prompt given to the LLM is a concatenation of the instruction, statically and dynamically selected in-context examples, and the sentence to be decomposed. The in-context decomposition examples used in our approach ($\mathcal{D}_{\text{R-ND}}$) are based on Russellian and neo-Davidsonian theories (§5).

and then aggregate results to give a single score or label for the entire claim. Since each subclaim is evaluated, we get a localized view of which parts of the claim are supported. The more atomic the subclaims are, the more precisely we can localize the information in the claim that differs from a trusted reference. Since these approaches rely on decomposition, the better the decomposition method the more reliable the results.

FACTSCORE (Min et al., 2023) measures factual precision of model-generated text with respect to a knowledge source. A generated passage is split into sentences, which are decomposed into subclaims by an LLM. The percentage of subclaims supported by a retrieved knowledge source (e.g., Wikipedia excerpts) is the FACTSCORE for the passage. FAITHSCORE (Jing et al., 2023) takes a similar approach for evaluating the outputs of vision-language models, in which the knowledge source against which the subclaims are evaluated is an image. They additionally require that the subclaims fit into certain domain-specific categories such as color and count.

The WICE dataset (Kamoi et al., 2023) contains annotations for whether subclaims in human-written text are supported, partially supported, or not supported by external reference documents, from which claim-level support labels are derived. Kamoi et al. (2023) also apply their LLM-based Claim-Split approach to entailment classification, in which entailment scores for each subclaim are aggregated to give an entailment score for the whole claim.

3 Evaluating Decomposition Quality

Previous work on evaluating the veracity of generated text attributes the resulting score to the quality of the generation, overlooking the role of metric's decomposition step. However, higher quality decompositions mean that we can more reliably measure the quality of the generation. Depending on the characteristics of the decomposition method (e.g., how atomic its decompositions are), a metric like FACTSCORE can change for the same underlying generated text (Figure 2). Furthermore, FACTSCORE implicitly assumes complete and coherent decompositions. However, the decomposition step can introduce unclaimed information or omit existing (possibly incorrect) claims, which introduces measurement error into FACTSCORE.

3.1 Qualitative Evaluation

What makes a decomposition higher quality? The subclaims must be faithful to the original claim. In other words, they must cohere with (are supported or entailed by) the original claim.[3] To be of

[3] In contrast to the coherence theory of truth, the correspondence theory deems a statement to be true if it matches a situation in reality. It is not in the purview of a decomposition model to determine whether a claim agrees with a knowledge source; that is the purpose of the validator. In other words,

the greatest use for localizing discrepancies with a trusted reference, the subclaims should cover all parts of the claim and also be as atomic as possible. Different methods decompose claims to various degrees, with some methods producing more or fewer subclaims. We explore these various characteristics across decomposition methods in §8.1.

3.2 Quantitative Evaluation: DecompScore

We develop a measure of decomposition method quality by utilizing the same procedure as FActScore, namely using an LLM to assign a binary judgment of support for every subclaim. Rather than providing an external knowledge source as context for the validator, we provide the original sentence that was decomposed, thus identifying the subclaims that are supported by the original sentence.

The DecompScore of a decomposition method is the average number of supported subclaims per passage produced by that decomposition method. This metric indicates which method generates the most subclaims that cohere with the sentence being decomposed. For example, if a text is decomposed into a large number of subclaims but DecompScore is low, we can infer that the subclaims produced by the decomposition method are not of good quality. The optimal value of DecompScore for a particular passage is difficult to determine because we do not have a set of reference decompositions, but in general, methods that produce decompositions with high atomicity and coverage will achieve higher DecompScore.

Entailment is another notion of coherence that could be used to evaluate whether a subclaim is a valid part of the decomposition. In practice, we find high correlation (Figure 7 in Appendix C) between DecompScore and the average number of subclaims entailed by the original claim using a strong natural language inference (NLI) model (Nie et al., 2020).[4]

4 Methods of Claim Decomposition

We study three types of claim decomposition methods, which are discussed below.

4.1 LLM prompting

Much of the recent work for claim decomposition utilizes a prompted LLM-based method, typically with in-context example decompositions (Min et al., 2023; Kamoi et al., 2023; Chen et al., 2023c; Jing et al., 2023; Mohri and Hashimoto, 2024). The in-context examples can be dynamically selected using a retrieval model (Min et al., 2023). We use three instructions from prior work (Min et al., 2023; Chen et al., 2023c; Kamoi et al., 2023) and one of our own, with various static and retrieved in-context examples. Notably, our approach uses manually decomposed in-context examples based on philosophical and linguistic theories, which are discussed in §5. The approaches' configurations are outlined in Table 1.

The LLM prompting approach is flexible and unstructured, allowing for the generation of arbitrary text. This text generation nature of LLMs produces fluent natural language decompositions by incorporating words outside the original sentence (in contrast to, e.g., PropSegmEnt (Chen et al., 2023b)), but this also permits hallucinations and forces us to relinquish control over the model's outputs due to the large output space. We can adapt the instructions and in-context examples to encourage certain characteristics in the output (such as coherence and atomicity), but ultimately there is no mechanism to guarantee they are reflected in the output. However, in-context examples that are dynamically chosen based on high similarity with the claim to be decomposed could encourage similar styles of decomposition, which may provide some amount of controllability. A simple prompt-in, subclaims-out interface also avoids issues of parsing into and generating out of an explicit intermediate semantic representation, designing such a representation in the first place, and overcoming any structural weaknesses in such a representation.

4.2 Shallow semantic parsing

Rather than relying on an LLM for the decomposition, we can use a more structured analysis of the text. We use PredPatt (White et al., 2016; Zhang et al., 2017), a rule-based system for extracting predicate-argument sub-structures from a syntactic dependency parse. We take these sub-structures

the validator is the "fact checker". A validator that appeals to a knowledge source is actually following a coherence theory of truth (where the given set of statements is the information contained in the knowledge source). The validator's adherence to a coherence theory of truth is apparent if we consider a case in which the subclaims are not grounded in reality but rather derived from a work of fiction. We can judge a statement like "Sherlock Holmes lives at 221B Baker Street" to be true even though it is false in reality.

[4] https://huggingface.co/ynie/roberta-large-snli_mnli_fever_anli_R1_R2_R3-nli

as representing the propositional content of subclaims. Goyal and Durrett (2020) use similar intuitions about a correspondence between syntactic dependency arcs and semantic units to decompose a claim based on arcs in a dependency parse.

The resulting subclaims contain only words from the original sentence, and are often not grammatical sentences.[5] The subclaims in a valid decomposition should be full sentences in order to be validated by DecompScore and FActScore, and for this reason, we use an LLM (`gpt-3.5-turbo-instruct`) to convert the PredPatt outputs into fluent, natural language. Details are given in Appendix B. Although the resulting strings are often full grammatical sentences, the LLM does not guarantee this behavior.[6]

4.3 LLM prompting with parse

Combining syntactic structure with the flexibility of text generation could support a more grounded decomposition from an LLM. We use an LLM prompting method, but this time supplied with a parsed version of the original sentence. We use Trankit (Nguyen et al., 2021), a state-of-the-art dependency parser, to obtain dependency parses (Zeman et al., 2019) (in the CoNLL-U format) of each claim as well as each in-context learning example. Because CoNLL-U formatted parses (Nivre et al., 2017) are token-heavy, fewer in-context examples are provided. Prompt details can be found in Table 1.

This method inherits the fluency and flexibility of LLM prompting while grounding the LLM's response in a syntactic analysis, resulting in (hopefully) a higher quality decomposition. While we hope the added structure imposes controllability, LLMs can still generate subclaims that do not cohere with the original claim.

5 Russellian and Neo-Davidsonian decomposition

The notion of claim decomposition has roots in the philosophical literature. We draw inspiration from Bertrand Russell's theory of logical atomism for how claims should be decomposed into their atomic components.

Russell defines atomic facts as properties of individuals or relations between individuals from which all other facts are composed (Russell, 1918b).[7, 8] We take individuals to be entities and eventualities mentioned in the sentence. This kind of Russellian analysis accords with neo-Davidsonian analysis (Castañeda, 1967; Parsons, 1990) (building on Davidson (1967)), in which the logical form of a sentence is decomposed fully to a conjunction of unary predicates (akin to properties of individuals) and binary predicates (akin to relations between individuals).

We manually decompose the 21 in-context examples from Min et al. (2023) into lists of such Russellian atomic propositions that we further decompose following neo-Davidsonian intuitions into unary and binary relations to obtain the smallest units that are claimed in each sentence: each subclaim designates a property of an individual or a relation between two individuals.[9] Our decompositions are listed in Table 10. These in-context examples are retrieved in the same way as the examples are retrieved for the FActScore prompt.

6 Data

We use the released data from Min et al. (2023), which consists of biographies of 500 individuals generated from each of 12 LMs (following their notation, we call the text generation models LM_{SUBJ}).[10] We do not modify the biographies generated by Min et al. (2023), nor do we generate

[5] PredPatt can add short strings like "is/are" and "poss" to indicate being and possession, respectively, but these additions do not make the propositions fluent.

[6] A model for determining grammatical acceptability could be included in this approach to filter out ungrammatical strings or send them back for rewriting (Warstadt et al., 2019).

[7] Ludwig Wittgenstein theorizes a similar idea of elementary propositions that assert atomic "states of affairs". On the whole, we find Wittgenstein's theory to be less actionable than Russell's. Incidentally, Wittgenstein later abandoned this theory in part due to the color exclusion problem, which we avoid by not requiring independence of subclaims, instead requiring only that each subclaim is claimed by the sentence.

[8] For Russell, "facts" are "the kind of thing that makes a proposition true or false" (Russell, 1918a), and for Wittgenstein they are states of affairs. In both cases, they are not propositions but rather conditions of the world. Russell and Wittgenstein use the terms "atomic proposition" and "elementary proposition", respectively, to refer to the corresponding truth function or expression of an atomic fact. The NLP literature uses the term "atomic fact" to mean the corresponding proposition, typically written in natural language.

[9] We do not include existence as a property of entities. Consider the sentences: "Allan Pinkerton was a detective who worked in the United States." and "Sherlock Holmes was a detective who worked in London." From just the sentences alone and without external knowledge, there is no way to tell that one of these people existed and one didn't.

[10] GPT-4 (OpenAI, 2023); ChatGPT; InstructGPT; Alpaca 7B, 13B, 65B (Taori et al., 2023); Vicuna 7B, 13B (Chiang et al., 2023); Dolly 12B (Biderman et al., 2023); StableLM-tuned-alpha 7B (Taori et al., 2023; Chiang et al., 2023; Anand et al., 2023); Oasst-pythia 12B; and MPT Chat 7B.

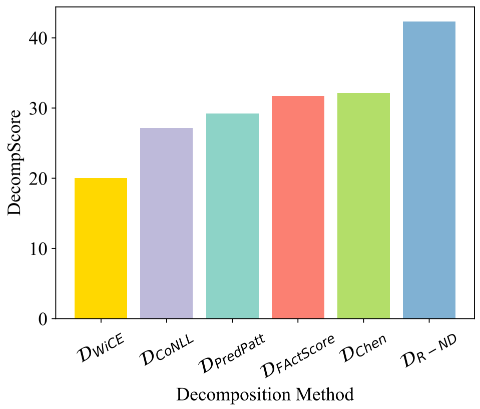

Figure 3: DECOMPSCORE (macro-averaged across LM_{SUBJ}) of different decomposition methods. A higher DECOMPSCORE is better.

additional ones. We treat them as static documents to investigate various decomposition methods applied to the sentences in the biographies.

7 Experiments

We use the data described in §6 for sentence-level decomposition with the methods outlined in §4 and §5. Model specifications are listed in Appendix B. We evaluate using DECOMPSCORE with Inst-LLAMA from Min et al. (2023) (LLAMA trained on Super Natural Instructions (Wang et al., 2022; Touvron et al., 2023)) and FACTSCORE with the Inst-LLAMA + retrieval + NPM setting. In total, generating decompositions took 120 GPU-hours, computing DECOMPSCORE took 250 GPU-hours, and computing FACTSCORE took 450 GPU-hours, all using a Quadro RTX 6000.

8 Results

DECOMPSCORE results are shown in Figure 3, with full results in Table 2 (Appendix A). $\mathcal{D}_{\text{R-ND}}$ attains the highest DECOMPSCORE (i.e., highest average number of supported subclaims per biography) with 42.3, followed by $\mathcal{D}_{\text{Chen et al.}}$ and $\mathcal{D}_{\text{FACTSCORE}}$, both with around 32. $\mathcal{D}_{\text{WICE}}$ produces the fewest average supported subclaims, with a DECOMPSCORE of 20.0, less than half that of $\mathcal{D}_{\text{R-ND}}$. The DECOMPSCOREs of $\mathcal{D}_{\text{PredPatt}}$ and $\mathcal{D}_{\text{CoNLL-U}}$ fall between $\mathcal{D}_{\text{WICE}}$ and $\mathcal{D}_{\text{FACTSCORE}}$, with $\mathcal{D}_{\text{PredPatt}}$ achieving a slightly higher DECOMPSCORE (29.2) than $\mathcal{D}_{\text{CoNLL-U}}$ (27.1).

FACTSCORE results are shown in Figure 2, with full results in Table 4 and Figure 4 (Appendix A). Undesirably, the FACTSCORE values vary based on the decomposition method used.

8.1 Qualitative Analysis

We analyze all decomposition methods on two sentences generated by GPT-4: one about Alfred Hitchcock and one about John Nash.[11] The decompositions, alongside our own manual decompositions, are shown in Table 8 and Table 9 in Appendix D. The evaluation criteria we use are coherence to the original sentence, coverage of the information claimed, and atomicity.

We observe that for the sentence about Alfred Hitchcock (Table 8), no decomposition method separates the date into month, day, and year or the location into city and state. No method generates the subclaim "Alfred Hitchcock passed away", opting to always include the date or location. Additionally, no method outputs all four combinations arising from the conjunction of "captivate" and "inspire" with "audiences" and "filmmakers". $\mathcal{D}_{\text{R-ND}}$ is the only method to separate "suspenseful" from "thrilling"; every other method keeps them as one unit. Similarly, many methods keep "captivate and inspire" as one unit; $\mathcal{D}_{\text{R-ND}}$ and $\mathcal{D}_{\text{FACTSCORE}}$ are the only ones to always split this conjunction.

We see that for the sentence about John Nash (Table 9), $\mathcal{D}_{\text{R-ND}}$, $\mathcal{D}_{\text{FACTSCORE}}$, and $\mathcal{D}_{\text{Chen et al.}}$ all output a large number of subclaims. However, many of the subclaims generated by $\mathcal{D}_{\text{FACTSCORE}}$ and $\mathcal{D}_{\text{Chen et al.}}$ incrementally add information to their other subclaims, which makes them non-atomic. This behavior of incrementally adding information can be expected given that it occurs in the in-context examples used by Min et al. (2023). This incrementality makes it more difficult to localize errors in the original claim because the textual support of the new information in the subclaim undesirably depends on the re-used information also being supported. All methods except for $\mathcal{D}_{\text{WICE}}$ generate non-atomic subclaims that combine Nash's bachelor's and master's degrees. $\mathcal{D}_{\text{R-ND}}$, $\mathcal{D}_{\text{CoNLL-U}}$, and $\mathcal{D}_{\text{PredPatt}}$ mention the degrees without the additional information that they were for mathematics, which increases atomicity; the other methods describe them always as "degree[s] in mathematics".

In our experiments, $\mathcal{D}_{\text{FACTSCORE}}$ and $\mathcal{D}_{\text{Chen et al.}}$ use the same in-context examples with slightly dif-

[11] "Alfred Hitchcock passed away on April 29, 1980, in Bel-Air, California, leaving behind a rich legacy of suspenseful and thrilling films that continue to captivate and inspire audiences and filmmakers alike." and "Nash demonstrated a natural aptitude for mathematics from a young age and earned his bachelor's and master's degrees in mathematics from the Carnegie Institute of Technology (now Carnegie Mellon University) in 1948."

ferent instructions and generate similar decompositions on the two sentences (identical decompositions on the Nash sentence). This behavior suggests that the in-context examples influence the decomposition more than the instruction does.

Takeaways For both sentences, we observe that many subclaims in our manual decompositions are missed by the decomposition methods, but the methods with the most coverage are $\mathcal{D}_{\text{R-ND}}$, $\mathcal{D}_{\text{Chen et al.}}$, $\mathcal{D}_{\text{FActScore}}$, and $\mathcal{D}_{\text{WiCE}}$. All methods but $\mathcal{D}_{\text{PredPatt}}$ have perfect coherence for both sentences. In general, we observe that $\mathcal{D}_{\text{WiCE}}$ has low atomicity,[12] as does $\mathcal{D}_{\text{CoNLL-U}}$ because it does not split conjunctions. $\mathcal{D}_{\text{PredPatt}}$ exhibits many issues: its subclaims are not atomic, often not fluent (despite using an LLM to make them more fluent), and not coherent with the original claim (e.g., "The bachelor possessed a master's degree").

8.2 Quantitative Analysis

Even though all decomposition methods are run on the same set of static biographies, they differ in FActScore and number of subclaims generated (averaged over LM_{SUBJ}: Figure 2, per LM_{SUBJ}: Table 4). This finding indicates that FActScore is sensitive to the method of decomposition that is used. The most reliable estimate of the generated text's "true" factual precision is the FActScore achieved by the highest quality decomposition method.

We hypothesize that $\mathcal{D}_{\text{PredPatt}}$'s FActScore is low because it produces subclaims not likely to be supported by the external knowledge source,[13] while also being constrained to using only the words in the sentence and missing implicit subclaims not extractable as predicate-argument structures from the dependency parse. Additionally, only 86% of the subclaims it produces are supported by the original claim (Table 6 in Appendix A), which agrees with our previous observation that its outputs have low coherence.

$\mathcal{D}_{\text{FActScore}}$ and $\mathcal{D}_{\text{Chen et al.}}$ both achieve a DecompScore around 32, and since they use the same in-context examples in our experiments, this further suggests that the decompositions are robust to the wording of the instruction in the prompt. Additionally, the similarity of the configuration of $\mathcal{D}_{\text{R-ND}}$ to those of $\mathcal{D}_{\text{FActScore}}$ and $\mathcal{D}_{\text{Chen et al.}}$ suggests that it is the manually decomposed in-context examples used in $\mathcal{D}_{\text{R-ND}}$ that are responsible for its higher DecompScore.

Because the in-context examples seem to have a larger effect on the decompositions than the instructions do and because we provide fewer examples in $\mathcal{D}_{\text{CoNLL-U}}$ due to the large token count of the parses, we evaluate the effect on decomposition of the number of in-context examples given. We use the same prompt specifications as in $\mathcal{D}_{\text{FActScore}}$ in Table 1, but use the same number of static examples as in $\mathcal{D}_{\text{CoNLL-U}}$ (one). We find that using fewer examples produces around the same number of subclaims (+1.3 subclaims on average), and achieves similar DecompScore (-0.69%) and FActScore (+0.06%). Overall, using fewer in-context examples does not appear to have much impact on either decomposition quality or factual precision.

When evaluating FActScore on only the *supported* subclaims (as determined in the calculation of DecompScore), in most cases, this subset of subclaims yields a higher FActScore (Table 4, Table 5, Figure 4, Figure 5 in Appendix A),[14] indicating that subclaims which do not cohere with the original sentence are likely also not supported by the knowledge source. Although simple, this filtering step removes potential errors introduced during decomposition. The fewest amount of subclaims (0.2 on average) are removed from $\mathcal{D}_{\text{WiCE}}$'s decompositions (compare Table 2 and Table 3 in Appendix A), indicating very high coherence, and the most are removed from $\mathcal{D}_{\text{PredPatt}}$'s decompositions (4 subclaims per biography on average), suggesting low coherence to the original sentence. On average, 1.2 out of 43.5 subclaims are removed from $\mathcal{D}_{\text{R-ND}}$'s decompositions.

To ensure that decompositions have high coherence, we recommend that subclaims produced by a decomposition method that are not supported by the original claim be filtered out (giving full coherence by construction). In doing so, unclaimed information that is introduced during the decomposition step is removed and not incorrectly attributed back to the generated text being evaluated.

Takeaways Despite $\mathcal{D}_{\text{WiCE}}$ having high coherence and coverage, it has the lowest Decomp-

[12] The instructions given to annotators for evaluating WiCE's Claim-Split decomposition method include an example that explicitly states that one of its subclaims can be further decomposed but to ignore that issue, which suggests atomicity is not prioritized in that method.

[13] For example, the mention of "civil rights" results in the subclaim "Rights are civil", which is likely not explicitly asserted in the retrieved Wikipedia passages.

[14] There are 4 exceptions out of 84 cases, and the maximum decrease in FActScore is 0.2%.

SCORE because it has low atomicity, which makes it undesirable as a decomposition method for use in a localized textual support metric.

Achieving a higher FACTSCORE with a particular decomposition method does not necessarily mean the decompositions are also of high quality. Although $\mathcal{D}_{\text{R-ND}}$ achieves lower FACTSCORE than most of the other methods, it has a far higher DECOMPSCORE than the other methods, which we hypothesize is due to our manually decomposed in-context examples. Such a method that produces a large number of supported subclaims that (qualitatively) have high coverage and atomicity is far more favorable in the textual support evaluation setting because it increases confidence in the results obtained from the downstream metric.

9 Related Work

Evaluation We evaluated decomposition methods that produce subclaims in sentential natural language, primarily by using contemporary technologies like large language models (§4). We review other methods of decomposition used in evaluation of textual support here.

Question answering (Wang et al., 2020; Durmus et al., 2020; Scialom et al., 2021; Fabbri et al., 2022) has been used for evaluating abstractive summarization. These methods generally ask questions only about noun phrases, require generating questions (the decomposition step), and require extracting answer spans, after which (typically lexical) heuristics determine if the answers between the summary and reference agree. Higher decomposition quality in this paradigm would involve generating a large number of highly focused questions, which would give better localized coverage of the claims made in the summary.

Goodrich et al. (2019) evaluate summarization by extracting relation tuples from a model-generated summary which are compared to relations extracted from a ground-truth summary. Fan et al. (2023) improve upon this approach by extracting fact tuples using semantic role labeling. Goyal and Durrett (2020) evaluate the factuality of model-generated text by obtaining entailment labels on each arc in a dependency parse, which assumes a correspondence between syntactic dependency arcs and semantic units (the same core assumption made by PredPatt).

In addition to evaluating *whether* text is supported, there has also been work on evaluating types of textual errors (Pagnoni et al., 2021; Devaraj et al., 2022; Mishra et al., 2024) and evaluating ambiguously supported claims (Glockner et al., 2024). Although designed to be used at the sentence-level, such methodologies can also be applied to subclaims. For further discussion about identifying and mitigating errors in model-generated text, such as hallucinations, we refer the reader to Ji et al. (2023) and Ye et al. (2023).

NLI Decomposition is also used for sub-sentence level NLI. PROPSEGMENT (Chen et al., 2023b) identifies subclaims by marking tokens in a claim that are part of the subclaim. They use propositional-level NLI to detect hallucinations by comparing tokens in entailed and non-entailed propositions. Sub-sentence entailment judgments can also be combined to make sentence-level or paragraph-level entailment judgments more interpretable and robust (Stacey et al., 2022, 2023; Kamoi et al., 2023).

Fact Verification Verifying the accuracy of statements depends on high quality decompositions to facilitate evidence retrieval. Chen et al. (2023a) build a system for complex claim verification by generating lists of yes/no questions that align to specific aspects of a claim. Chen et al. (2022) build a similar system that also asks implied subquestions. Li et al. (2023) and Milbauer et al. (2023) align generated claims with statements in documents that entail or contradict the claim. Similarly, Ernst et al. (2021) align propositions between reference summaries and source documents—which is similar to the fact verification task. A model trained on their dataset was later used to cluster propositions in a system for multi-document summarization (Ernst et al., 2022). Chen et al. (2023c) use decomposition to find matching subclaims ("atomic propositions") across sentences to train proposition-level representations using contrastive learning. The proposition representations are used for retrieving propositions from a corpus that support a given proposition.

10 Conclusion

We observe that a downstream metric of textual support, namely factual precision as measured by FACTSCORE, is sensitive to the method it uses to decompose a claim into its subclaims. This finding leads us to measure decomposition quality using our proposed metric DECOMPSCORE so that we can use the most appropriate decomposition method among those we consider.

We show that an LLM prompted with in-context learning examples that we manually decompose by following intuitions from logical atomism and neo-Davidsonian semantics outperforms other methods. Decompositions generated by our method contain the greatest number of subclaims supported by the original claim among the methods we consider. Qualitative analysis and comparison to manual decompositions demonstrate that all the decomposition methods we consider still miss subclaims and many generate non-atomic subclaims, indicating there still remains room for improvement.

Limitations

Metrics like FACTSCORE and DECOMPSCORE are able to evaluate only information that is claimed in a generated text. Information relevant to an upstream query may be absent in the text, whether accidentally or intentionally, and these evaluation approaches cannot account for that.

This study is limited to the domain of entity biographies, so it is not representative of all use cases. Additionally, the data is monolingual (English), and we do not know if these results hold across other languages.

Running LLMs can be expensive. Because of this, we chose to use LLAMA instead of ChatGPT as the validator, but even running that model is not financially feasible for everyone to use.

Ethics Statement

LLMs are well-known to hallucinate information, and mitigation of hallucination is still an active area of research. Using LLMs to decompose a claim into subclaims can introduce new factual errors. Despite attempts to remove such errors (for example, by filtering out subclaims that are not supported by the original claim according to DECOMPSCORE), errors can still persist. Caution must be taken when relying on text generated from a model.

Acknowledgements

We thank Nathaniel Weir, Kate Sanders, and Anqi Liu for helpful discussions, as well as the anonymous reviewers for their feedback. This work has been supported in part by the U.S. National Science Foundation under grant No. 2204926.

References

Yuvanesh Anand, Zach Nussbaum, Brandon Duderstadt, Benjamin Schmidt, and Andriy Mulyar. 2023. Gpt4all: Training an assistant-style chatbot with large scale data distillation from gpt-3.5-turbo. https://github.com/nomic-ai/gpt4all.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, Usvsn Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 2397–2430. PMLR.

Héctor-Neri Castañeda. 1967. Comments on D. Davidson's 'The logical form of action sentences'. In Nicholas Rescher, editor, *The Logic of Decision and Action*, pages 104–112. University of Pittsburgh Press.

Jifan Chen, Grace Kim, Aniruddh Sriram, Greg Durrett, and Eunsol Choi. 2023a. Complex claim verification with evidence retrieved in the wild. *arXiv preprint arXiv:2305.11859*.

Jifan Chen, Aniruddh Sriram, Eunsol Choi, and Greg Durrett. 2022. Generating literal and implied subquestions to fact-check complex claims. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3495–3516, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Sihao Chen, Senaka Buthpitiya, Alex Fabrikant, Dan Roth, and Tal Schuster. 2023b. PropSegmEnt: A large-scale corpus for proposition-level segmentation and entailment recognition. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8874–8893, Toronto, Canada. Association for Computational Linguistics.

Sihao Chen, Hongming Zhang, Tong Chen, Ben Zhou, Wenhao Yu, Dian Yu, Baolin Peng, Hongwei Wang, Dan Roth, and Dong Yu. 2023c. Sub-sentence encoder: Contrastive learning of propositional semantic representations. *arXiv preprint arXiv:2311.04335*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.

Donald Davidson. 1967. The logical form of action sentences. In Nicholas Rescher, editor, *The Logic of Decision and Action*, pages 81–95. University of Pittsburgh Press.

Ashwin Devaraj, William Sheffield, Byron Wallace, and Junyi Jessy Li. 2022. Evaluating factuality in text simplification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7331–7345, Dublin, Ireland. Association for Computational Linguistics.

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5055–5070, Online. Association for Computational Linguistics.

Ori Ernst, Avi Caciularu, Ori Shapira, Ramakanth Pasunuru, Mohit Bansal, Jacob Goldberger, and Ido Dagan. 2022. Proposition-level clustering for multi-document summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1765–1779, Seattle, United States. Association for Computational Linguistics.

Ori Ernst, Ori Shapira, Ramakanth Pasunuru, Michael Lepioshkin, Jacob Goldberger, Mohit Bansal, and Ido Dagan. 2021. Summary-source proposition-level alignment: Task, datasets and supervised baseline. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 310–322, Online. Association for Computational Linguistics.

Alexander Fabbri, Chien-Sheng Wu, Wenhao Liu, and Caiming Xiong. 2022. QAFactEval: Improved QA-based factual consistency evaluation for summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2587–2601, Seattle, United States. Association for Computational Linguistics.

Jing Fan, Dennis Aumiller, and Michael Gertz. 2023. Evaluating factual consistency of texts with semantic role labeling. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 89–100, Toronto, Canada. Association for Computational Linguistics.

Max Glockner, Ieva Staliūnaitė, James Thorne, Gisela Vallejo, Andreas Vlachos, and Iryna Gurevych. 2024. AmbiFC: Fact-Checking Ambiguous Claims with Evidence. *Transactions of the Association for Computational Linguistics*, 12:1–18.

Ben Goodrich, Vinay Rao, Peter J. Liu, and Mohammad Saleh. 2019. Assessing the factual accuracy of generated text. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, page 166–175, New York, NY, USA. Association for Computing Machinery.

Tanya Goyal and Greg Durrett. 2020. Evaluating factuality in generation with dependency-level entailment. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3592–3603, Online. Association for Computational Linguistics.

Andrew Hickl and Jeremy Bensley. 2007. A discourse commitment-based framework for recognizing textual entailment. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 171–176, Prague. Association for Computational Linguistics.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Liqiang Jing, Ruosen Li, Yunmo Chen, Mengzhao Jia, and Xinya Du. 2023. Faithscore: Evaluating hallucinations in large vision-language models.

Ryo Kamoi, Tanya Goyal, Juan Diego Rodriguez, and Greg Durrett. 2023. WiCE: Real-world entailment for claims in Wikipedia. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7561–7583, Singapore. Association for Computational Linguistics.

Miaoran Li, Baolin Peng, and Zhu Zhang. 2023. Self-checker: Plug-and-play modules for fact-checking with large language models. *arXiv preprint arXiv:2305.14623*.

Jeremiah Milbauer, Ziqi Ding, Zhijin Wu, and Tongshuang Wu. 2023. NewsSense: Reference-free verification via cross-document comparison. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 422–430, Singapore. Association for Computational Linguistics.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Abhika Mishra, Akari Asai, Vidhisha Balachandran, Yizhong Wang, Graham Neubig, Yulia Tsvetkov, and Hannaneh Hajishirzi. 2024. Fine-grained hallucination detection and editing for language models. *arXiv preprint arXiv:2401.06855*.

Christopher Mohri and Tatsunori Hashimoto. 2024. Language models with conformal factuality guarantees. *arXiv preprint arXiv:2402.10978*.

Minh Van Nguyen, Viet Dac Lai, Amir Pouran Ben Veyseh, and Thien Huu Nguyen. 2021. Trankit: A light-weight transformer-based toolkit for multilingual natural language processing. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 80–90, Online. Association for Computational Linguistics.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Joakim Nivre, Daniel Zeman, Filip Ginter, and Francis Tyers. 2017. Universal Dependencies. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Tutorial Abstracts*, Valencia, Spain. Association for Computational Linguistics.

OpenAI. 2023. Gpt-4 technical report.

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. 2021. Understanding factuality in abstractive summarization with FRANK: A benchmark for factuality metrics. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4812–4829, Online. Association for Computational Linguistics.

Terence Parsons. 1990. *Events in the Semantics of English: A Study in Subatomic Semantics*. MIT Press.

Bertrand Russell. 1918a. The philosophy of logical atomism, lecture 1. *The Monist*, 28.

Bertrand Russell. 1918b. The philosophy of logical atomism, lecture 2. *The Monist*, 28.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, Alex Wang, and Patrick Gallinari. 2021. QuestEval: Summarization asks for fact-based evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6594–6604, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Joe Stacey, Pasquale Minervini, Haim Dubossarsky, Oana-Maria Camburu, and Marek Rei. 2023. Logical reasoning for natural language inference using generated facts as atoms. *arXiv preprint arXiv:2305.13214*.

Joe Stacey, Pasquale Minervini, Haim Dubossarsky, and Marek Rei. 2022. Logical reasoning with span-level predictions for interpretable and robust NLI models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3809–3823, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Liyan Tang, Philippe Laban, and Greg Durrett. 2024. Minicheck: Efficient fact-checking of llms on grounding documents. *arXiv preprint arXiv:2404.10774*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Aaron Steven White, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2016. Universal Decompositional Semantics on Universal Dependencies. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1713–1723, Austin, Texas. Association for Computational Linguistics.

Hongbin Ye, Tong Liu, Aijia Zhang, Wei Hua, and Weiqiang Jia. 2023. Cognitive mirage: A review of hallucinations in large language models. *arXiv preprint arXiv:2309.06794*.

Daniel Zeman, Joakim Nivre, Mitchell Abrams, Noëmi Aepli, Željko Agić, Lars Ahrenberg, Gabrielė Aleksandravičiūtė, Lene Antonsen, Katya Aplonova, Maria Jesus Aranzabe, Gashaw Arutie, Masayuki Asahara, Luma Ateyah, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, Elena Badmaeva, Miguel Ballesteros, Esha Banerjee, Sebastian Bank, Verginica Barbu Mititelu, Victoria Basmov, Colin Batchelor, John Bauer, Sandra Bellato, Kepa Bengoetxea, Yevgeni Berzak, Irshad Ahmad Bhat, Riyaz Ahmad Bhat, Erica

Biagetti, Eckhard Bick, Agnė Bielinskienė, Rogier Blokland, Victoria Bobicev, Loïc Boizou, Emanuel Borges Völker, Carl Börstell, Cristina Bosco, Gosse Bouma, Sam Bowman, Adriane Boyd, Kristina Brokaitė, Aljoscha Burchardt, Marie Candito, Bernard Caron, Gauthier Caron, Tatiana Cavalcanti, Gülşen Cebiroğlu Eryiğit, Flavio Massimiliano Cecchini, Giuseppe G. A. Celano, Slavomír Čéplö, Savas Cetin, Fabricio Chalub, Jinho Choi, Yongseok Cho, Jayeol Chun, Alessandra T. Cignarella, Silvie Cinková, Aurélie Collomb, Çağrı Çöltekin, Miriam Connor, Marine Courtin, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Elvis de Souza, Arantza Diaz de Ilarraza, Carly Dickerson, Bamba Dione, Peter Dirix, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Hanne Eckhoff, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Olga Erina, Tomaž Erjavec, Aline Etienne, Wograine Evelyn, Richárd Farkas, Hector Fernandez Alcalde, Jennifer Foster, Cláudia Freitas, Kazunori Fujita, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Moa Gärdenfors, Sebastian Garza, Kim Gerdes, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta González Saavedra, Bernadeta Griciūtė, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Nizar Habash, Jan Hajič, Jan Hajič jr., Mika Hämäläinen, Linh Hà Mỹ, Na-Rae Han, Kim Harris, Dag Haug, Johannes Heinecke, Felix Hennig, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Jena Hwang, Takumi Ikeda, Radu Ion, Elena Irimia, Ọlájídé Ishola, Tomáš Jelínek, Anders Johannsen, Fredrik Jørgensen, Markus Juutinen, Hüner Kaşıkara, Andre Kaasen, Nadezhda Kabaeva, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Boris Katz, Tolga Kayadelen, Jessica Kenney, Václava Kettnerová, Jesse Kirchner, Elena Klementieva, Arne Köhn, Kamil Kopacewicz, Natalia Kotsyba, Jolanta Kovalevskaitė, Simon Krek, Sookyoung Kwak, Veronika Laippala, Lorenzo Lambertino, Lucia Lam, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Ph01AF01A1ng Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Cheuk Ying Li, Josie Li, Keying Li, KyungTae Lim, Maria Liovina, Yuan Li, Nikola Ljubešić, Olga Loginova, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Héctor Martínez Alonso, André Martins, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Sarah McGuinness, Gustavo Mendonça, Niko Miekka, Margarita Misirpashayeva, Anna Missilä, Cătălin Mititelu, Maria Mitrofan, Yusuke Miyao, Simonetta Montemagni, Amir More, Laura Moreno Romero, Keiko Sophie Mori, Tomohiko Morioka, Shinsuke Mori, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Robert Munro, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, L01AF01A1ng Nguyễn Thị, Huyền Nguyễn Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima Nitisaroj, Hanna Nurmi, Stina Ojala, Atul Kr. Ojha, Adédayọ00D2 Olúòkun, Mai Omura, Petya Osenova, Robert Östling, Lilja Øvrelid, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Angelika Peljak-Łapińska, Siyao Peng, Cenel-Augusto Perez, Guy Perrier, Daria Petrova, Slav Petrov, Jason Phelan, Jussi Piitulainen, Tommi A Pirinen, Emily Pitler, Barbara Plank, Thierry Poibeau, Larisa Ponomareva, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Tiina Puolakainen, Sampo Pyysalo, Peng Qi, Andriela Rääbis, Alexandre Rademaker, Loganathan Ramasamy, Taraka Rama, Carlos Ramisch, Vinit Ravishankar, Livy Real, Siva Reddy, Georg Rehm, Ivan Riabov, Michael Rießler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Luisa Rocha, Mykhailo Romanenko, Rudolf Rosa, Davide Rovati, Valentin Roșca, Olga Rudina, Jack Rueter, Shoval Sadde, Benoît Sagot, Shadi Saleh, Alessio Salomoni, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Dage Särg, Baiba Saulīte, Yanin Sawanakunanon, Nathan Schneider, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Mo Shen, Atsuko Shimada, Hiroyuki Shirasu, Muh Shohibussirri, Dmitry Sichinava, Aline Silveira, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Isabela Soares-Bastos, Carolyn Spadine, Antonio Stella, Milan Straka, Jana Strnadová, Alane Suhr, Umut Sulubacak, Shingo Suzuki, Zsolt Szántó, Dima Taji, Yuta Takahashi, Fabio Tamburini, Takaaki Tanaka, Isabelle Tellier, Guillaume Thomas, Liisi Torga, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Andrius Utka, Sowmya Vajjala, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Eric Villemonte de la Clergerie, Veronika Vincze, Lars Wallin, Abigail Walsh, Jing Xian Wang, Jonathan North Washington, Maximilan Wendt, Seyi Williams, Mats Wirén, Christian Wittern, Tsegay Woldemariam, Tak-sum Wong, Alina Wróblewska, Mary Yako, Naoki Yamazaki, Chunxiao Yan, Koichi Yasuoka, Marat M. Yavrumyan, Zhuoran Yu, Zdeněk Žabokrtský, Amir Zeldes, Manying Zhang, and Hanzhi Zhu. 2019. Universal dependencies 2.5. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Sheng Zhang, Rachel Rudinger, and Benjamin Van Durme. 2017. An evaluation of PredPatt and open IE via stage 1 semantic role labeling. In *Proceedings of the 12th International Conference on Computational Semantics (IWCS) — Short papers*.

A Full Results

FACTSCORE evaluation is outlined in §2, and full results are reported in Table 4 and Figure 4. DECOMPSCORE evaluation is discussed in §3.2, and full results are reported in Table 2. Unlike FACTSCORE, we do not impose a length penalty in DECOMPSCORE because shorter passages naturally contain fewer subclaims. Percentages of subclaims that are judged to be supported (i.e., the coherence of each method) are shown in Table 6 and Figure 6.

FACTSCORE results based on the subclaims judged to cohere with the original claim (based on judgments obtained when computing DECOMPSCORE) are shown in Table 5 and Figure 5. The average numbers of subclaims per biography are reported in Table 3, and the average numbers of supported subclaims (i.e., the DECOMPSCORE) are reported in Table 2.

It is important to note the special cases and conditions placed on these results:

- The released data from Min et al. (2023) includes uninformative LM responses (e.g. "I'm sorry, I don't have any information on a person named...") Including these generations is valuable for evaluating factuality of a language model, however results in noise when evaluating decomposition quality. These uninformative responses are still processed by the decomposition methods we wish to evaluate, however the quality of decomposition is unaffected.

- Different language models are trained on different versions of Wikipedia, which introduces inconsistencies from the Wikipedia context used for fact-checking. This can affect FACTSCORE but does not affect DECOMPSCORE because it does not make use of external knowledge sources.

B Model Details

To reduce cost using the `text-davinci-003` model used by Min et al. (2023), we instead use InstructGPT (`gpt-3.5-turbo-instruct`) as the LLM for decomposition with 4K token context window, 512 `max_tokens` and a temperature of 0.7. This model costs $0.0015 per 1K input tokens and $0.0020 per 1K output tokens. `gpt-3.5-turbo-instruct` achieves Pearson correlation coefficients of over 0.97 for both FACTSCORE and number of subclaims generated compared to results reported by Min et al. (2023) (Table 7).

Inst-LLAMA is LLAMA trained on Super Natural Instructions (Wang et al., 2022; Touvron et al., 2023), and is used for all FACTSCORE and DECOMPSCORE evaluations. We use `max_sequence_length` of 2048 and `max_output_length` of 128.

For $\mathcal{D}_{\text{PredPatt}}$, we use `Trankit` for generating the dependency parse for each sentence. This parse is then used by `PredPatt` with the following flags: relative clauses, appositional modifiers, adjectival modifiers, conjunction, possessives, borrow_arg_for_relcl and strip all set to True, with the remaining flags (simple, cut, and big_args) set to False. We use `PredPatt` with Universal Dependencies v2.

We use `gpt-3.5-turbo-instruct` with the settings enumerated above for converting PredPatt outputs into natural language sentences with the following prompt:

Please turn my input utterances into a grammatically correct natural English sentence by resolving tense, fixing grammatical errors, and reordering words without changing meanings. Your output should not contain "is/are" or "poss". Your output should contain no hallucinated information and no redundant sentences. Just the modified utterance.

Input: born 1908 community leader
Output: The community leader was born in 1908.

Input: date of death is/are unknown
Output: The date of death is unknown.

Input: was an African - American social worker activist
Output: They were an African-American social worker activist.

Input: <subclaim>
Output:

When a prompt in the $\mathcal{D}_{\text{CoNLL-U}}$ approach exceeds the length allowed for the context window, examples are incrementally removed until the prompt fits. When a zero-shot prompt (no in-context examples) exceeds the size of the context window, we backoff and set the entire original sentence as the subclaim. In practice, we backoff 0.05% of the time: across 6000 passages (500 passages generated by each of 12 LM_{SUBJ}), twice we use one example and once we use the original sentence. We leave it to future work to reduce the size of the parses used in the prompt.

			DecompScore				
LM_{SUBJ}	$\mathcal{D}_{\mathcal{D}_{\text{R-ND}}}$	$\mathcal{D}_{\text{Chen}}$	$\mathcal{D}_{\text{WiCE}}$	$\mathcal{D}_{\text{FS}}$	$\mathcal{D}_{\text{FS2}}$	$\mathcal{D}_{\text{CoNLL}}$	$\mathcal{D}_{\text{PP}}$
Alpaca 7B	**21.9**	17.7	11.2	17.2	18.8	15.4	15.2
Alpaca 13B	**21.6**	16.9	10.5	16.5	18.2	15.0	14.9
Alpaca 65B	**21.9**	17.3	10.8	16.7	18.5	15.2	14.8
ChatGPT	**43.0**	32.5	20.2	32.4	33.9	27.3	29.0
Dolly 12B	**32.1**	24.9	15.2	24.3	26.8	21.9	20.5
GPT4	**76.0**	57.5	35.9	57.2	58.5	47.0	54.8
InstructGPT	**35.5**	27.6	17.2	26.9	28.8	23.4	23.1
MPT-Chat 7B	**47.7**	36.5	22.7	35.9	37.4	30.2	33.1
Oasst-pythia 12B	**56.7**	41.6	25.4	40.9	42.3	34.8	39.7
StableLM 7B	**38.2**	29.5	18.9	29.3	30.6	25.5	28.1
Vicuna 7B	**58.4**	43.8	27.4	43.4	45.4	36.7	41.1
Vicuna 13B	**54.6**	39.8	24.9	39.9	41.5	33.1	36.2
Macro-average	**42.3**	32.1	20.0	31.7	33.4	27.1	29.2

Table 2: DecompScore for each decomposition method and LM_{SUBJ}. Average number of subclaims generated per biography that are determined to be supported by the original sentence.

			# Subclaims				
LM_{SUBJ}	$\mathcal{D}_{\mathcal{D}_{\text{R-ND}}}$	$\mathcal{D}_{\text{Chen}}$	$\mathcal{D}_{\text{WiCE}}$	$\mathcal{D}_{\text{FS}}$	$\mathcal{D}_{\text{FS2}}$	$\mathcal{D}_{\text{CoNLL}}$	$\mathcal{D}_{\text{PP}}$
Alpaca 7B	**22.2**	17.9	11.3	17.3	19.0	15.7	16.4
Alpaca 13B	**22.0**	17.2	10.6	16.6	18.4	15.3	16.2
Alpaca 65B	**22.2**	17.5	10.9	16.9	18.6	15.5	16.0
ChatGPT	**44.2**	33.0	20.4	33.0	34.6	28.5	33.2
Dolly 12B	**33.0**	25.2	15.4	24.7	27.2	22.9	23.4
GPT4	**77.7**	58.2	36.2	57.9	59.2	48.6	63.6
InstructGPT	**36.3**	27.9	17.3	27.2	29.1	23.9	25.6
MPT-Chat 7B	**49.0**	37.0	22.9	36.3	37.8	31.1	37.4
Oasst-pythia 12B	**57.7**	41.8	25.5	41.2	42.6	35.4	44.6
StableLM 7B	**40.4**	30.7	19.4	30.4	32.0	27.4	33.4
Vicuna 7B	**59.8**	44.3	27.6	43.9	45.9	37.7	46.3
Vicuna 13B	**57.3**	44.6	25.1	45.8	42.8	34.8	42.2
Macro-average	**43.5**	32.9	20.2	32.6	33.9	28.1	33.2

Table 3: Average number of subclaims generated per biography.

	FActScore (%)						
$\mathrm{LM}_{\mathrm{SUBJ}}$	$\mathcal{D}_{\mathcal{D}_{\text{R-ND}}}$	$\mathcal{D}_{\text{Chen}}$	$\mathcal{D}_{\text{WICE}}$	$\mathcal{D}_{\text{FS}}$	$\mathcal{D}_{\text{FS2}}$	$\mathcal{D}_{\text{CoNLL}}$	$\mathcal{D}_{\text{PP}}$
Alpaca 7B	35.0	36.9	33.7	36.9	37.5	34.9	27.4
Alpaca 13B	38.9	40.3	35.1	40.8	41.1	38.3	30.0
Alpaca 65B	44.0	47.0	42.8	46.9	47.3	45.0	36.5
ChatGPT	48.2	52.1	51.4	52.2	52.2	50.7	36.8
Dolly 12B	16.5	16.3	13.9	16.7	17.2	15.5	10.4
GPT4	51.1	56.1	54.8	55.9	54.9	53.3	35.6
InstructGPT	40.1	43.2	43.2	43.6	43.4	41.7	31.5
MPT-Chat 7B	24.8	25.9	24.4	26.2	25.2	25.1	16.1
Oasst-pythia 12B	20.1	20.8	19.2	21.2	21.1	20.5	11.7
StableLM 7B	13.8	13.1	11.6	13.5	13.4	13.3	8.2
Vicuna 7B	32.4	34.5	34.0	35.2	34.9	33.8	21.7
Vicuna 13B	31.1	32.8	31.8	34.1	35.7	33.1	23.3
Macro-average	33.0	34.9	33.0	35.3	35.3	33.8	24.1

Table 4: FActScore of biographies generated by each $\mathrm{LM}_{\mathrm{SUBJ}}$ and decomposed with each method. Note: For evaluating decomposition quality, a larger FActScore is not necessarily better; we care about high confidence that FActScore is correct.

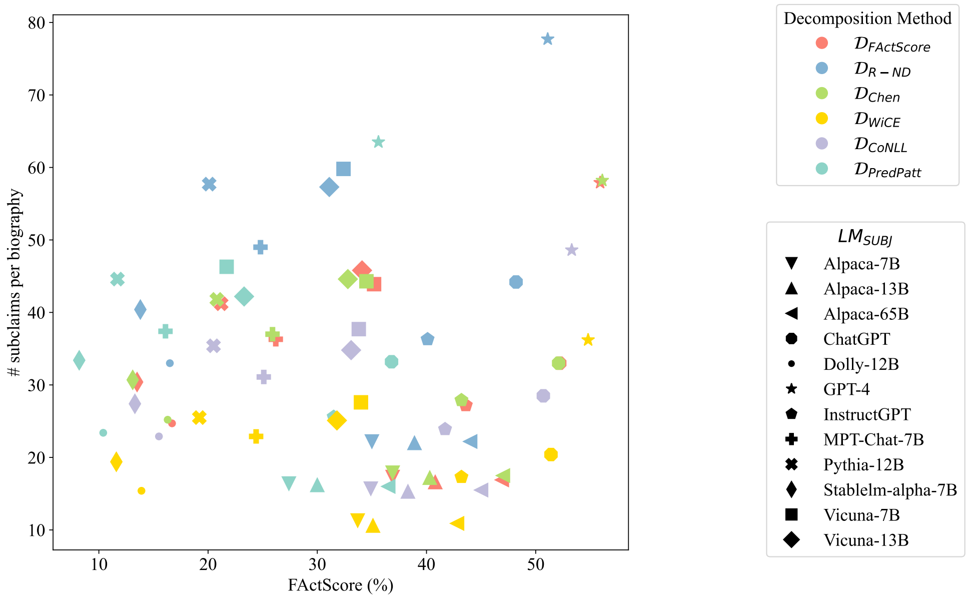

Figure 4: FActScore results for all claim decomposition methods and $\mathrm{LM}_{\mathrm{SUBJ}}$.

FActScore (%) After Filtering Out Unsupported Subclaims							
LM_{SUBJ}	$\mathcal{D}_{\mathcal{D}_{\text{R-ND}}}$	$\mathcal{D}_{\text{Chen}}$	$\mathcal{D}_{\text{WiCE}}$	$\mathcal{D}_{\text{FS}}$	$\mathcal{D}_{\text{FS2}}$	$\mathcal{D}_{\text{CoNLL}}$	$\mathcal{D}_{\text{PP}}$
Alpaca 7B	34.9	36.7	36.1	36.8	37.6	35.8	29.1
Alpaca 13B	40.1	40.8	40.2	41.4	41.2	39.9	31.3
Alpaca 65B	45.0	48.4	47.0	47.6	47.9	46.3	39.4
ChatGPT	55.8	60.5	60.2	59.9	59.9	59.1	45.1
Dolly 12B	17.1	17.1	16.1	17.6	17.7	16.9	12.2
GPT4	57.0	62.6	61.4	62.0	61.0	59.9	43.8
InstructGPT	40.7	43.5	43.6	44.0	44.0	42.6	34.3
MPT-Chat 7B	27.0	28.3	27.5	28.7	27.6	28.0	19.5
Oasst-pythia 12B	20.4	21.2	20.2	21.4	21.4	21.0	12.8
StableLM 7B	16.0	15.6	14.6	16.0	15.8	15.9	8.9
Vicuna 7B	35.7	38.6	38.4	38.8	38.4	37.6	25.3
Vicuna 13B	37.7	41.7	41.3	41.7	41.1	40.6	29.3
Macro-average	35.6	37.9	37.2	38.0	37.8	37.0	27.6

Table 5: FActScore of biographies after filtering out subclaims determined to be not supported by the original sentence (using DecompScore judgments). Note: For evaluating decomposition quality, a larger FActScore is not necessarily better; we care about high confidence that FActScore is correct.

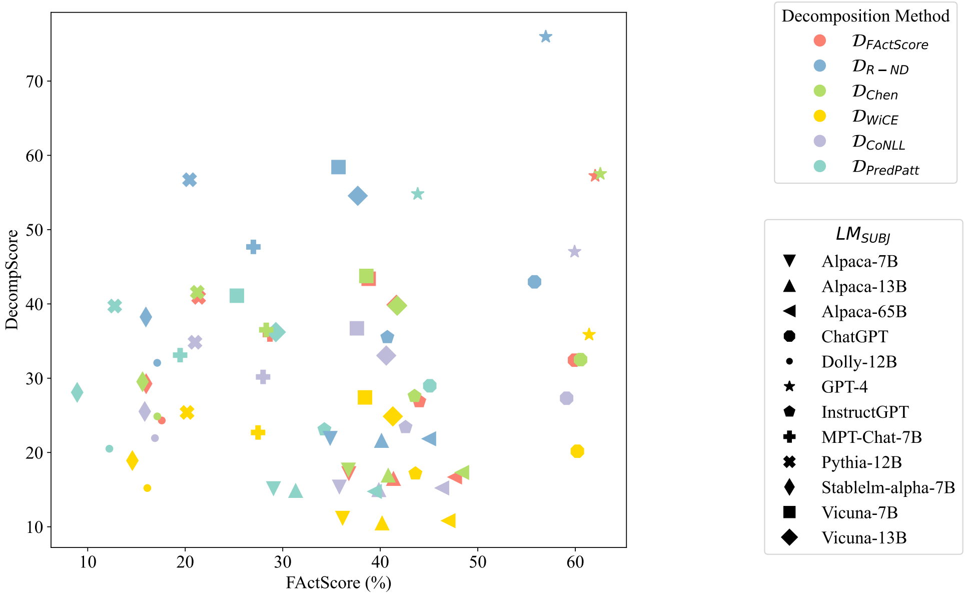

Figure 5: FActScore results after filtering out subclaims determined to be not supported by the original sentence (using DecompScore judgments) for all claim decomposition methods and LM_{SUBJ}.

	% Subclaims Supported						
LM$_{\text{SUBJ}}$	$\mathcal{D}_{\mathcal{D}_{\text{R-ND}}}$	$\mathcal{D}_{\text{Chen}}$	$\mathcal{D}_{\text{WICE}}$	$\mathcal{D}_{\text{FS}}$	$\mathcal{D}_{\text{FS2}}$	$\mathcal{D}_{\text{CoNLL}}$	$\mathcal{D}_{\text{PP}}$
Alpaca 7B	98.7	98.9	**99.2**	99.1	99.1	98.4	93.6
Alpaca 13B	98.6	99.0	**99.4**	99.0	99.2	98.2	93.2
Alpaca 65B	98.6	99.3	**99.4**	99.2	99.3	98.5	93.7
ChatGPT	93.0	95.9	96.7	**99.4**	94.5	89.0	80.0
Dolly 12B	97.4	98.7	**99.0**	98.7	98.6	96.5	89.6
GPT4	96.2	97.4	**98.3**	97.4	97.2	94.2	83.2
InstructGPT	98.1	99.1	**99.3**	99.0	99.0	98.0	90.8
MPT-Chat 7B	96.5	97.6	**98.4**	97.6	97.8	95.4	86.9
Oasst-pythia 12B	98.3	99.3	**99.4**	99.3	99.3	98.4	89.4
StableLM 7B	89.2	90.7	**94.1**	90.5	89.4	84.8	74.4
Vicuna 7B	94.8	97.0	**98.1**	96.3	96.5	92.9	84.1
Vicuna 13B	88.9	93.3	**95.4**	90.8	88.1	82.6	72.6
Macro-average	96.0	97.2	**98.1**	97.2	96.5	93.9	86.0

Table 6: Percentage of subclaims from each decomposition method and LM$_{\text{SUBJ}}$ that are judged to be supported by (cohere with) the original claim.

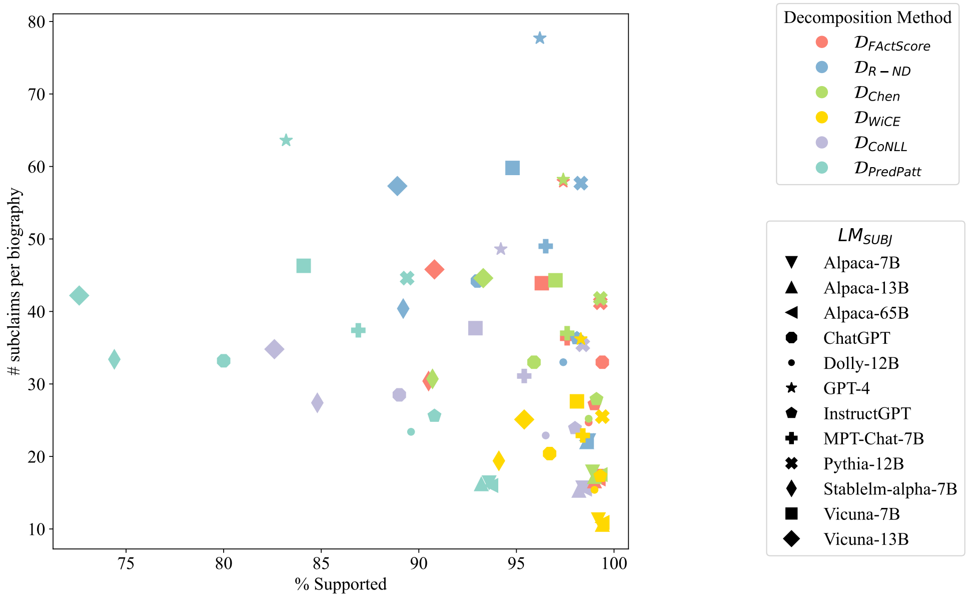

Figure 6: Percentage of subclaims that are supported by (cohere with) the original claim.

	FACTSCORE	Reported FACTSCORE	# subclaims	Reported # subclaims
Alpaca 7B	36.9	36.5	17.3	17.4
Alpaca 13B	40.8	40.3	16.6	16.6
Alpaca 65B	46.9	46.3	16.9	17.1
ChatGPT	52.2	60.4	33.0	37.0
Dolly 12B	16.7	17.1	24.7	24.6
GPT4	55.9	59.9	57.9	60.8
InstructGPT	43.6	41.7	27.2	27.7
MPT-Chat 7B	26.2	27.9	36.3	37.3
Oasst-pythia 12B	21.2	20.8	41.2	39.7
StableLM 7B	13.5	16.3	30.4	38.0
Vicuna 7B	35.2	36.9	43.9	45.6
Vicuna 13B	34.1	40.7	45.8	50.9
ρ	0.9786		0.9821	

Table 7: Pearson correlation coefficients (ρ) between our setup for computing FACTSCORE (using `gpt-3.5-turbo-instruct` for subclaim generation) and results reported by Min et al. (2023) (using `text-davinci-003` for subclaim generation).

C NLI Entailment

The numbers of subclaims that are judged to be entailed by the original sentence are highly correlated with the numbers of subclaims judged by an LLM to be supported by the original sentence (DECOMPSCORE), achieving a Pearson correlation coefficient of 0.9978 (Figure 7).

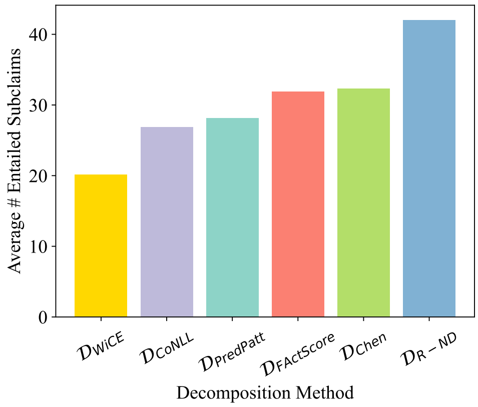

Figure 7: Average number of subclaims per passage that are entailed by their original sentential claim, as determined by an NLI model (Nie et al., 2020). Values are macro-averaged across LM_{SUBJ}.

D Decomposition Examples

We include examples of two sentences decomposed manually and by all claim decomposition methods evaluated. Table 8 contains the decompositions for the sentence “Alfred Hitchcock passed away on April 29, 1980, in Bel-Air, California, leaving behind a rich legacy of suspenseful and thrilling films that continue to captivate and inspire audiences and filmmakers alike." Table 9 contains the decompositions for the sentence “Nash demonstrated a natural aptitude for mathematics from a young age and earned his bachelor’s and master’s degrees in mathematics from the Carnegie Institute of Technology (now Carnegie Mellon University) in 1948.”

E Russellian/Neo-Davidsonian In-context Learning Examples

The manually decomposed sentences used as in-context examples for $\mathcal{D}_{\text{R-ND}}$ are shown in Table 10.

Manual	- Alfred Hitchcock passed away.
	- Alfred Hitchcock's death occurred in April.
	- Alfred Hitchcock's death occurred on the 29th day of a month.
	- Alfred Hitchcock's death occurred in 1980.
	- Alfred Hitchcock's death occurred in Bel-Air.
	- Bel-Air is in California.
	- Alfred Hitchcock left behind a legacy.
	- Alfred Hitchcock's legacy is rich.
	- Alfred Hitchcock's legacy contains films.
	- The films in Alfred Hitchcock's legacy are suspenseful.
	- The films in Alfred Hitchcock's legacy are thrilling.
	- The films in Alfred Hitchcock's legacy captivate audiences.
	- The films in Alfred Hitchcock's legacy captivate filmmakers.
	- The films in Alfred Hitchcock's legacy inspire audiences.
	- The films in Alfred Hitchcock's legacy inspire filmmakers.
$\mathcal{D}_{\text{R-ND}}$	- Alfred Hitchcock passed away on April 29, 1980.
	- His death occurred in Bel-Air, California.
	- Alfred Hitchcock had a legacy.
	- Alfred Hitchcock's legacy contains suspenseful films.
	- Alfred Hitchcock's legacy contains thrilling films.
	- Alfred Hitchcock's films continue to captivate audiences.
	- Alfred Hitchcock's films continue to inspire filmmakers.
	- Alfred Hitchcock left behind his legacy.
$\mathcal{D}_{\text{Chen et al.}}$	- Alfred Hitchcock passed away on April 29, 1980.
	- Alfred Hitchcock passed away in Bel-Air, California.
	- Alfred Hitchcock left behind a rich legacy.
	- Alfred Hitchcock's legacy includes suspenseful and thrilling films.
	- Alfred Hitchcock's films continue to captivate and inspire audiences.
	- Alfred Hitchcock's films continue to captivate and inspire filmmakers.
$\mathcal{D}_{\text{WiCE}}$	- Alfred Hitchcock passed away on April 29, 1980.
	- He passed away in Bel-Air, California.
	- He left behind a rich legacy of suspenseful and thrilling films.
	- His films continue to captivate and inspire audiences.
	- His films continue to inspire filmmakers.
$\mathcal{D}_{\text{FActScore}}$	- Alfred Hitchcock passed away on April 29, 1980.
	- Alfred Hitchcock passed away in Bel-Air, California.
	- Alfred Hitchcock left behind a legacy of films.
	- Alfred Hitchcock's films were suspenseful and thrilling.
	- Alfred Hitchcock's films continue to captivate audiences.
	- Alfred Hitchcock's films continue to inspire filmmakers.
$\mathcal{D}_{\text{CoNLL-U}}$	- Alfred Hitchcock passed away on April 29, 1980.
	- He passed away in Bel-Air, California.
	- He left behind a rich legacy.
	- His legacy includes suspenseful and thrilling films.
	- His films continue to captivate and inspire audiences and filmmakers.
$\mathcal{D}_{\text{PredPatt}}$	- Alfred Hitchcock passed away on April 29 , 1980 , in Bel - Air.
	- Bel - Air is/are in California
	- Alfred Hitchcock left behind a rich legacy of suspenseful and thrilling films that continue to captivate and inspire audiences and filmmakers alike
	- The legacy of suspenseful and thrilling films that continue to captivate and inspire audiences and filmmakers alike is/are rich
	- The films that continue to captivate and inspire audiences and filmmakers alike are suspenseful
	- Suspenseful and thrilling films continue to captivate audiences.
	- Suspenseful and thrilling films continue to captivate filmmakers alike.
	- That continue to inspire.

Table 8: Decomposition comparison for the sentence "Alfred Hitchcock passed away on April 29, 1980, in Bel-Air, California, leaving behind a rich legacy of suspenseful and thrilling films that continue to captivate and inspire audiences and filmmakers alike."

Manual	- Nash demonstrated an aptitude for mathematics. - Nash's aptitude for mathematics was natural. - Nash's demonstration of an aptitude for mathematics began at a young age. - Nash earned his bachelor's degree. - Nash's bachelor's degree is in mathematics. - Nash's bachelor's degree is from the Carnegie Institute of Technology. - Nash's bachelor's degree was earned in 1948. - Nash earned his master's degree. - Nash's master's degree is in mathematics. - Nash's master's degree is from the Carnegie Institute of Technology. - Nash's master's degree was earned in 1948. - The Carnegie Institute of Technology is now Carnegie Mellon University.
$\mathcal{D}_{\text{R-ND}}$	- Nash demonstrated a natural aptitude for mathematics. - His natural aptitude for mathematics was evident from a young age. - He earned a bachelor's degree in mathematics. - He earned a master's degree in mathematics. - He earned his degrees from the Carnegie Institute of Technology. - The Carnegie Institute of Technology is now known as Carnegie Mellon University. - He earned his degrees in 1948. - He earned his bachelor's degree in mathematics in 1948. - He earned his master's degree in mathematics in 1948.
$\mathcal{D}_{\text{Chen et al.}}$	- Nash demonstrated a natural aptitude for mathematics. - Nash demonstrated a natural aptitude for mathematics from a young age. - Nash earned his bachelor's degree in mathematics. - Nash earned his master's degree in mathematics. - Nash earned his bachelor's and master's degrees in mathematics. - Nash earned his bachelor's and master's degrees in mathematics from the Carnegie Institute of Technology. - The Carnegie Institute of Technology is now Carnegie Mellon University. - Nash earned his bachelor's and master's degrees in mathematics from the Carnegie Institute of Technology (now Carnegie Mellon University). - Nash earned his bachelor's and master's degrees in mathematics from the Carnegie Institute of Technology (now Carnegie Mellon University) in 1948.
$\mathcal{D}_{\text{WiCE}}$	- Nash demonstrated a natural aptitude for mathematics from a young age. - Nash earned his bachelor's degree in mathematics from the Carnegie Institute of Technology in 1948. - Nash earned his master's degree in mathematics from the Carnegie Institute of Technology in 1948. - The Carnegie Institute of Technology is now known as Carnegie Mellon University.
$\mathcal{D}_{\text{FActScore}}$	- Nash demonstrated a natural aptitude for mathematics. - Nash demonstrated a natural aptitude for mathematics from a young age. - Nash earned his bachelor's degree in mathematics. - Nash earned his master's degree in mathematics. - Nash earned his bachelor's and master's degrees in mathematics. - Nash earned his bachelor's and master's degrees in mathematics from the Carnegie Institute of Technology. - Carnegie Institute of Technology is now called Carnegie Mellon University. - Nash earned his bachelor's and master's degrees in mathematics from the Carnegie Institute of Technology (now Carnegie Mellon University). - Nash earned his bachelor's and master's degrees in mathematics from the Carnegie Institute of Technology (now Carnegie Mellon University) in 1948.
$\mathcal{D}_{\text{CoNLL-U}}$	- Nash demonstrated an aptitude for mathematics. - Nash earned his bachelor's and master's degrees in mathematics. - Nash's degrees were from Carnegie Institute of Technology. - The institute is now known as Carnegie Mellon University. - Nash received his degrees in 1948.
$\mathcal{D}_{\text{PredPatt}}$	- Nash demonstrated a natural aptitude for mathematics from a young age. - Aptitude for mathematics is natural. - They were young. - Nash earned his bachelor 's and master 's degrees in mathematics from the Carnegie Institute of Technology in 1948. - He had a bachelor 's and master 's degrees in mathematics. - The bachelor possessed a master's degree. - The Carnegie Institute of Technology is now Carnegie Mellon University.

Table 9: Decomposition comparison for the sentence "Nash demonstrated a natural aptitude for mathematics from a young age and earned his bachelor's and master's degrees in mathematics from the Carnegie Institute of Technology (now Carnegie Mellon University) in 1948."

He made his acting debut in the film The Moon is the Sun’s Dream (1992), and continued to appear in small and supporting roles throughout the 1990s.

- He has an acting debut.
- He acted in a film.
- His acting debut was in a film.
- His acting debut was in The Moon is the Sun’s Dream.
- He acted in The Moon is the Sun’s Dream.
- The Moon is the Sun’s Dream is a film.
- The Moon is the Sun’s Dream was released in 1992.
- His acting debut occurred in 1992.
- He appeared in small roles.
- He appeared in supporting roles.
- His small roles occurred throughout the 1990s.
- His supporting roles occurred throughout the 1990s.
- His appearance in small roles occurred after his acting debut.
- His appearance in supporting roles occurred after his acting debut.

He is also a successful producer and engineer, having worked with a wide variety of artists, including Willie Nelson, Tim McGraw, and Taylor Swift.

- He is a producer.
- He is successful at being a producer.
- He is an engineer.
- He is successful at being an engineer.
- He has worked with a wide variety of artists.
- Willie Nelson is an artist.
- He has worked with Willie Nelson.
- Tim McGraw is an artist.
- He has worked with Tim McGraw.
- Taylor Swift is an artist.
- He has worked with Taylor Swift.

In 1963, Collins became one of the third group of astronauts selected by NASA and he served as the back-up Command Module Pilot for the Gemini 7 mission.

- NASA selected a third group of astronauts.
- Collins belonged to the third group of astronauts.
- Collins was selected by NASA.
- Collins’s selection by NASA occurred in 1963.
- The Gemini 7 mission has a back-up Command Module Pilot.
- Collins’s role in the Gemini 7 mission was as the back-up Command Module Pilot.
- Collins participated in the Gemini 7 mission.

In addition to his acting roles, Bateman has written and directed two short films and is currently in development on his feature debut.

- Bateman has acting roles.
- Bateman has written short films.
- The number of short films Bateman has written is two.
- Bateman has directed short films.
- The number of short films Bateman has directed is two.
- Bateman is currently in development on his feature debut.
- The two short films were made before his feature debut.
- His acting roles came before his feature debut.

Michael Collins (born October 31, 1930) is a retired American astronaut and test pilot who was the Command Module Pilot for the Apollo 11 mission in 1969.

- Michael Collins was born in October.
- Michael Collins was born on the 31st day of a month.
- Michael Collins was born in 1930.
- Michael Collins is retired.
- Michael Collins is American.
- Michael Collins was an astronaut.
- Michael Collins was a test pilot.
- Michael Collins participated in the Apollo 11 mission.
- Michael Collins’s participation in the Apollo 11 mission occurred in 1969.
- The Apollo 11 mission was active in 1969.
- The day of Michael Collins’s birth occurred before his year of participation in the Apollo 11 mission.
- The Apollo 11 mission had a Command Module Pilot.
- Michael Collins’s role in the Apollo 11 mission was as the Command Module Pilot.

He was an American composer, conductor, and musical director.
- He was American.
- He was a composer.
- He was a conductor.
- He was a musical director.

She currently stars in the romantic comedy series, Love and Destiny, which premiered in 2019.
- She stars in Love and Destiny.
- Love and Destiny is a series.
- Love and Destiny is a romantic comedy.
- Love and Destiny premiered in 2019.

His music has been described as a mix of traditional Mexican and Latin American styles, as well as jazz, folk, and rock.
- He has music.
- His music has been described.
- His music has been described as a mix of styles.
- His music has been described as containing elements of traditional styles of music.
- His music has been described as containing elements of Mexican style of music.
- His music has been described as containing elements of Latin American style of music.
- His music has been described as containing elements of jazz music.
- His music has been described as containing elements of folk music.
- His music has been described as containing elements of rock music.

He also serves as an ambassador for the charity Leonard Cheshire Disability.
- He has a role in Leonard Cheshire Disability.
- His role in Leonard Cheshire Disability is as an ambassador.
- Leonard Cheshire Disability is a charity.

He began his career in Nashville in the late 1950s and has since released numerous albums, including a greatest hits collection in 1999.
- He has a career.
- His career began in Nashville.
- His career began in the late 1950s.
- He has released albums.
- His released albums are numerous.
- He released a collection.
- His collection contains greatest hits.
- His collection was released in 1999.
- The release of his albums occurred after he began his career.

He has been performing since the age of 8, when he joined a band in his hometown of Guadalajara and has since gone on to record six studio albums and several singles of his own original material.
- He has been performing.
- He started performing at the age of 8.
- He joined a band.
- He joined a band at the age of 8.
- His band was in Guadalajara.
- His hometown is Guadalajara.
- He has recorded studio albums.
- The number of studio albums he has recorded is six.
- He has recorded singles.
- He has several singles.
- His studio albums are his own original material.
- His singles are his own original material.
- His recording of studio albums occurred after he joined a band.
- His recording of singles occurred after he joined a band.

She is also the former President of the Malaysian Chinese Association (MCA) from 2010 to 2013.
- She had a role in the Malaysian Chinese Association.
- Her role in the Malaysian Chinese Association was as its President.
- Her tenure as President of the Malaysian Chinese Association started in 2010.
- Her tenure as President of the Malaysian Chinese Association ended in 2013.
- MCA is another name for the Malaysian Chinese Association.

During his professional career, McCoy played for the Broncos, the San Diego Chargers, the Minnesota Vikings, and the Jacksonville Jaguars.
- McCoy had a professional career.
- McCoy played for the Broncos.
- McCoy played for the San Diego Chargers.
- The Chargers are from San Diego.
- McCoy played for the Minnesota Vikings.
- The Vikings are from Minnesota.
- McCoy played for the Jacksonville Jaguars.
- The Jaguars are from Jacksonville.

Miller has been described as the architect of Trump's controversial immigration policies, and has previously worked for Alabama Senator Jeff Sessions on immigration issues. - Miller has been described. - Miller has been described as an architect. - Miller has been described as an architect of Trump's controversial immigration policies. - Trump has immigration policies. - Trump's immigration policies are controversial. - Miller worked for Jeff Sessions. - Jeff Sessions is a Senator. - Jeff Sessions represents Alabama. - Miller worked on immigration issues. - Miller's work for Jeff Sessions involved immigration issues.
Her work is often described as whimsical and dreamlike. - She has work. - Her work has been described. - Her work is described as whimsical. - Her work is described as dreamlike. - The description of her work as whimsical has occurred often. - The description of her work as dreamlike has occurred often.
He graduated from the United States Military Academy in 1952, and then went on to serve in the United States Air Force. - He graduated from the United States Military Academy. - His graduation from the United States Military Academy occurred in 1952. - He served in the United States Air Force. - His service in the United States Air Force occurred after his graduation from the United States Military Academy.
He is best known for his roles in the films Memories of Murder (2003), The Host (2006), (...) and Parasite (2019). - He had a role in Memories of Murder. - Memories of Murder is a film. - Memories of Murder was released in 2003. - He had a role in The Host. - The Host is a film. - The Host was released in 2006. - He had a role in Parasite. - Parasite is a film. - Parasite was released in 2009. - His role in Memories of Murder is one of his best known. - His role in The Host is one of his best known. - His role in Parasite is one of his best known.
Song Kang-ho was born in Gongju, South Korea in 1967. - Song Kang-ho was born. - Song Kang-ho's birth occurred in Gongju. - Song Kang-ho's birth occurred in South Korea. - Song Kang-ho's birth occurred in 1967. - Gongju is in South Korea.
He studied theater at Chung-Ang University in Seoul. - He studied. - He studied theater. - He studied at Chung-Ang University. - His study of theater occurred at Chung-Ang University. - Chung-Ang University is located in Seoul.
His breakthrough came with the leading role in the acclaimed crime-drama film Memories of Murder in 2003. - He had a breakthrough. - His breakthrough was based on a leading role. - His breakthrough was based on his role in Memories of Murder. - His breakthrough occurred in 2003. - He had a leading role. - He had a leading role in Memories of Murder. - Memories of Murder is a film. - The genre of Memories of Murder is crime-drama. - Memories of Murder is acclaimed. - Memories of Murder was released in 2003.
This was followed by the monster movie The Host in 2006, which became the highest-grossing film in Korean history at the time. - This was followed by The Host. - The Host is a movie. - The Host was released in 2006. - The genre of The Host is monster movie. - The Host became the highest-grossing film in Korean history.

Table 10: Manually decomposed examples used for in-context examples by $\mathcal{D}_{\text{R-ND}}$.