Statistical Deficiency for Task Inclusion Estimation

Loïc Fosse[1,2], Frédéric Béchet [2,4], Benoît Favre[2,3], Géraldine Damnati[1],
Gwénolé Lecorvé[1], Maxime Darrin[4,5,6,8], Philippe Formont[4,6,7,8], Pablo Piantanida[4,6,8,9]

[1]Orange Research, Lannion, France, [2]CNRS, LIS, Aix Marseille Université, France,
[3]CNRS, LIG, Univ. Grenoble Alpes, Grenoble, France,
[4]International Laboratory on Learning Systems (ILLS - IRL CNRS), Montréal,
[5]McGill University, [6]Mila - Quebec AI Institute, [7]ÉTS Montréal,
[8]Université Paris-Saclay, [9]CNRS, CentraleSupélec.

Contact: loic.fosse@orange.com

Abstract

Tasks are central in machine learning, as they are the most natural objects to assess the capabilities of current models. The trend is to build general models able to address any task. Even though transfer learning and multitask learning try to leverage the underlying task space, no well-founded tools are available to study its structure. This study proposes a theoretically grounded setup to define the notion of task and to compute the **inclusion** between two tasks from a statistical deficiency point of view. We propose a tractable proxy as information sufficiency to estimate the degree of inclusion between tasks, show its soundness on synthetic data, and use it to reconstruct empirically the classic NLP pipeline.[1]

1 Introduction

Having a well-defined set of tasks with known or assumed dependency relationships has historically been a key element in building or evaluating Natural Language Processing (NLP) systems. For instance, Named Entity Recognition (NER) and summarization are two well-established tasks for which annotated datasets exist, and it is commonly accepted that the summarization task, at least in the news domain, requires NER skills to be performed effectively. As a consequence, studying generated summaries from the perspective of retained named entities is a relevant evaluation angle (Pagnoni et al., 2021; Berezin and Batura, 2022; Akani et al., 2023). According to this principle, a more general hypothesis is that multi-task training (Caruana, 1997) provides cross-task generalization (Mishra et al., 2022; Ye, 2024; Wang et al., 2024; Baxter, 2000; Wu et al., 2020) since tasks share dependencies. However, with the rise of instruct-tuned models (Wei et al., 2021), tasks can be directly defined by prompting large language models. This dramatically increase of the addressable tasks' space makes the notions of ground truth and labeled dataset more fuzzy, and raises many questions about the capabilities of these seemingly all-powerful models. Notably, it is unclear how many tasks a single model can correctly handle, how many parameters are necessary to capture a given task, and what proportion of the model is dedicated to language understanding, task solving, and memorization.

In this paper, to advance the understanding of the notion of task and its manipulation within language models, we are interested in studying intrinsic relationships between tasks. Building on the intuition that some tasks are necessary conditions to others (*e.g.* NER is a necessary condition for summarization), we propose a framework to discover statistical *task inclusion relationships*. Potential application to the discovery of such relations is to build smaller, more compute-efficient datasets (Zamir et al., 2018) and, more generally, to build better data-mix when training models (Ye et al., 2024), or more orthogonal benchmarks. Our main approach will rely on statistical simulation methods (Le Cam, 1964, 1996) to decide whether one task can be transformed into another. While we theoretically show the shortcomings of naively measuring cross-task performance by directly applying each model to each other task, the contributions of the paper are:

- **A theoretical framework for task definition and inclusion.** Based on information concepts and theory, we propose a clear definition of a task and candidate notions of inclusion (independent of the notion of model).

- **An experimental setup for task comparison.** We propose a tractable proxy to measure task inclusion through statistical reductions.

- **An empirical rediscovery of the NLP pipeline** Experiments suggest that our framework reconstructs the expected partial order for a sample of

---

[1]Code and data are available here

linguistic tasks from the classical NLP pipeline.

2 Related work

Discovering the task space underlying structure (Turing, 1950; Winograd, 1987) is a common problem in machine learning (ML), to compare human and machine representations of tasks but more generally to leverage potential structure for more efficient learning (Li and Hiratani, 2025). We list the main families of task comparison methods and discuss how our new one differs from them.

Task similarity.   Evaluating similarity between tasks is one of the main area of interest, as it is the crux to discovering transfer learning or meta-learning opportunities (Schmidhuber, 1987; Zhou et al., 2021). It can be performed by comparing the same model trained on various tasks (Achille et al., 2019; Shui et al., 2019), by comparing the data distributions of different tasks (Ethayarajh et al., 2022). Unlike work on task similarity, we aim at finding a non-symmetrical notion of task inclusion. We discuss in App. H limitations of similarity based approaches and why the proposed setup is more suited for task relationship study.

Task merging.   Inspired from ensemble methods (Dietterich, 2000), model merging focuses on combining several existing models to create a new one. Recent methods either use arithmetic operations (Ilharco et al., 2023; Tao et al., 2024; Ortiz-Jimenez et al., 2024; Zhou et al., 2024; Zeng et al., 2025) or more complex aggregation methods (Yadav et al., 2023; Jin et al., 2023; Yang et al., 2023), with the goal of solving conflicts or interferences between models (Yu et al., 2020; Sener and Koltun, 2018) and thus tasks. While task and model merging focuses on the parameter space (whose dimension is excessively large), the tools developed here focus on the activation space. However, we propose in App. H an analysis of parameter space based approaches and we show some limitations, despite some interesting behaviors.

Task Transfer.   Transfer learning (Torrey and Shavlik, 2010; Hanneke and Kpotufe, 2024; Lange et al., 2021) consists in leveraging a model pretrained for a new task for a given task, either as an initialization point for further training or to generate useful representations. Although most of the time one uses a generic pre-trained model and trains directly for the new task, Vu et al. (2020) showed that some tasks might benefit from training on an intermediate task, effectively building a path of (easily) transferable tasks. In computer vision, this phenomenon has been studied and quantified by Bao et al. (2019). Zamir et al. (2018), obtained similar results showing connections between various visual tasks and were able to leverage these structures to optimize training of multitask models (Zhang and Yang, 2021). Knowledge transferability between different tasks is also at the heart of modern ML and generalization as exhibited by models such as T5 (Wei et al., 2021; Khashabi et al., 2020), or more recently instruct-type models (Zhang et al., 2023b) with multi-task training leading to significantly stronger results. Task transfer is mainly based on successive fine-tuning processes, as well as on the study of the fine-tuned models parameters geometry (Fisher information). The tools developed here focus on fine-tuned model activations enabling us to avoid certain learning costs and connect with powerful theoretical results.

Probing.   Understanding what is encoded in a model has been a question of interest which led to probing methods. It consists in assessing the activations of a model on a downstream task (Guillaume and Yoshua, 2017; Chen et al., 2021; Rogers et al., 2021; Wallat et al., 2023; Nikolaev and Padó, 2023; Zhao et al., 2024a; Waldis et al., 2024). Some studies assess encoder-type models with probing methods to understand what the model encodes after fine-tuning on a task (Durrani et al., 2021; Merchant et al., 2020; Mosbach et al., 2020), in order to understand how tasks impact pre-trained models. The main observation is that deepest layers will focus more on the task, while first layers remain generic. This result is confirmed in (Durrani et al., 2022) with the use of clustering methods. However, probing has some drawbacks as discussed in (Pimentel et al., 2020; Kunz and Kuhlmann, 2020), where it is clearly explained that it can lead to miss-interpretation. Discussion about probing interpretation is provided in Sec. C.1. Moreover, the last layer interpretation does not seem to hold for decoder-only generative language models (which we are using here) (Gromov et al., 2024), even when fine-tuned on a task (*c.f.* Sec. 5).

3 Theoretical framework

We introduce here the theoretical set-up we will be using, which is mainly a probabilistic one. After defining a task (Definition 1), we will start by proposing a strict probabilistic definition of task in-

clusion, and a relaxed definition (Definition 2) that states that a task is included in another one if the estimation process of the latter provides sufficient information on the former. Then we propose a way of computing this crucial notion of informativeness through statistical deficiency (Definition 3). Beyond this theoretically grounded approach, we finally propose a tractable way to estimate deficiency through information sufficiency. The approach is developed in the following subsections and can be synthesized in Figure 1.

Notations. We denote by $\mathcal{P}(\mathsf{X})$ the set of all probability measures on X. For any random variable $X \in \mathsf{X}$, we'll denote by $\mathbb{P}_X \in \mathcal{P}(\mathsf{X})$ the associated push-forward measure. Given two probability measures P and Q on the same space, we denote by $\|P - Q\|_{\mathrm{TV}}$, the total variation distance. Given some spaces X and Y, we denote by $\mathcal{M}(\mathsf{Y}|\mathsf{X})$ the set of all Markov Kernels from X to Y, which can be seen (under certain assumptions) as the set of all conditional probability measures $\mathbb{P}_{Y|X}$. Given $K \in \mathcal{M}(\mathsf{Z}|\mathsf{Y})$ and $M \in \mathcal{M}(\mathsf{Y}|\mathsf{X})$, we denote by $K \circ M \in \mathcal{M}(\mathsf{Z}|\mathsf{X})$ the composition of kernels[2].

3.1 Task and inclusion

Definition 1 (Task). Given input data $X \in \mathsf{X}$ and response $Y \in \mathsf{Y}$, a task is the joint probability measure $\mathbb{P}_{XY} \in \mathcal{P}(\mathsf{X} \times \mathsf{Y})$.

Remark 1. Definition 1 provides a simple generic definition of tasks in a ML context, which is (at least implicitly) adopted in many works (Maurer et al., 2016; Baxter, 2000).

Given Definition 1, many cases can appear while comparing tasks, yielding different interpretations. Two tasks can be considered on different input marginal distributions $\mathbb{P}_X$, which is known in the literature as the *domain shift* (Wortsman et al., 2022; Taori et al., 2020; Radford et al., 2021; Kumar et al., 2022). Tasks comparison in that case will lead to interpretation about tasks' domain. Our goal is different. We seek to compare tasks in terms of skills, which refers to the conditional measure $\mathbb{P}_{Y|X}$, *i.e.* the skills to estimate Y given X. To clarify, we make the following assumptions:

- (H1) All our tasks are probability measures on the same space $(\mathsf{X} \times \mathsf{Y})$ which is true in the generative paradigm where X and Y are both texts.

- (H2) For all tasks, the marginal distribution $\mathbb{P}_X$ will always remain the same. This is equivalent

to considering that all our tasks are performed on the same input text[3].

Given Definition 1 and the different hypotheses, solving a task will be considered as the estimation of the conditional probability measure $\mathbb{P}_{Y|X}$. Then, we define the inclusion between two tasks as the answer to the following question: Given two tasks $\mathbb{P}_{XY_U}$ and $\mathbb{P}_{XY_V}$, does the estimation of $\mathbb{P}_{Y_U|X}$ implies being able to estimate $\mathbb{P}_{Y_V|X}$? This question gives the simplest idea of the inclusion between two tasks: if we can perform one task, we can perform the other one. However, this is too restrictive since the estimation of $\mathbb{P}_{Y_U|X}$ can effectively not directly imply having the entire measure $\mathbb{P}_{Y_V|X}$. However, having $\mathbb{P}_{Y_U|X}$ can give strong hints (information) on the shape of $\mathbb{P}_{Y_V|X}$ (Boudiaf et al., 2021) and we would like to capture this situation as an inclusion one[4]. This is the reason why we define a relaxed version of the inclusion,

Definition 2 (Lenient-inclusion). Given two tasks $\mathbb{P}_{XY_U}$ and $\mathbb{P}_{XY_V}$, we say that $\mathbb{P}_{XY_V}$ is included into $\mathbb{P}_{XY_U}$ (denoted as $\mathbb{P}_{XY_V} \widetilde{\subset} \mathbb{P}_{XY_U}$), iff the estimation of $\mathbb{P}_{Y_U|X}$ is informative about $\mathbb{P}_{Y_V|X}$.

In Definition 2, the notion of informativeness depends on the context. The goal in the following will be to define different versions of the informativeness of one task about another. Definition 2 seems more simple and requires less constraints on the shapes of $\mathbb{P}_{Y_U|X}$ and $\mathbb{P}_{Y_V|X}$ and we'll stick to this definition. Moreover, we can refer to an intensity of the inclusion which refers to how informative one task is about another. Still, to find inclusion, we must be able to manipulate very complex probability measures which are here our tasks[5]. Thus, a meaningful and tractable representation of tasks is needed to address the inclusion estimation. Such a representation can be obtained by looking at a model fine-tuned to solve the task. In fact, it has been shown in (Boudiaf et al., 2021; Achille and Soatto, 2018; Tishby et al., 2000) that models fine-tuned to solve a task produce sufficient statistics of the task, in Fisher's sense (Keener, 2010, Definition 3.2 p.43)[6]. In the case of current language models, these sufficient statistics are essentially the continuous representations (or embeddings) $Z \in \mathsf{Z}$ of text X produced by the models. These embed-

---

[2]We give further details about the formalism in Sec. A.1.

[3]We provide more details about this hypothesis in Sec. B.1

[4]In Sec. C.1 we detail why we can have this situation

[5]In Sec. B.2 arguments are developed, with the only use of measure theory.
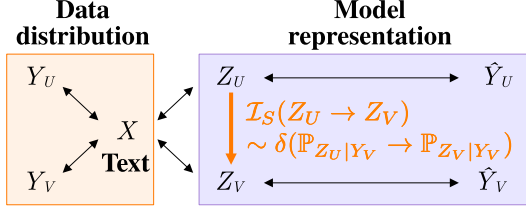
[6]For more details about this we refer to Sec. C.2

Figure 1: Illustration of proposed task comparison framework. X is textual input, Y is reference output, $\hat{Y}$ is system output, Z represents embeddings, (U, V) is a pair of tasks. $\delta()$ is statistical deficiency and $\mathcal{I}_S$ is the information sufficiency proxy.

dings will thus be used as proxies to estimate task inclusion. In the following, for sake of simplicity we will refer to $\mathbb{P}_{XY_U}$ and $\mathbb{P}_{XY_V}$ as respectively task U and task V.

3.2 Deficiency: a notion of inclusion

Definition 2 states that $V \widetilde{\subset} U$ if and only if solving the task U is informative about the task V. One way to verify this, is by comparing embeddings' distributions of models fine-tuned on respectively U (Z_U) and V (Z_V). In the following, we will use the notation $\mathbb{P}_{Z_U|Y_U} \in \mathcal{M}(\mathsf{Z}|\mathsf{Y})$. This kernel describes the distribution of Z_U given the values Y_U takes. A good kernel $\mathbb{P}_{Z_U|Y_U}$ would cluster the embeddings Z_U depending on the value Y_U can take (similar values of Y_U lead to similar Z_U), assuring we can infer Y_U from Z_U[7]. Then, following the path of inclusion of V into U, a question of interest would be: how good is the kernel $\mathbb{P}_{Z_U|Y_V}$? Or equivalently: how clustered Z_U is relatively to Y_V? Statistical deficiency, first introduced by Blackwell (1951, 1953) in the context of comparison of statistical experiments, provides a set up to compare $\mathbb{P}_{Z_U|Y_V}$ and $\mathbb{P}_{Z_V|Y_V}$ (the latter being considered good for task V), providing one possible answer to previous questions.

Definition 3 (Deficiency (Le Cam, 1964)). Let $\mathbb{P}_{XY_U}$ and $\mathbb{P}_{XY_V}$ be two tasks, and Z_U and Z_V the embeddings of X given by fine-tuned models. The deficiency $\delta(\mathbb{P}_{Z_U|Y_V} \to \mathbb{P}_{Z_V|Y_V})$ measures the informativeness of $\mathbb{P}_{Z_U|Y_V}$ about $\mathbb{P}_{Z_V|Y_V}$ and is defined as:

$$\delta(\mathbb{P}_{Z_U|Y_V} \to \mathbb{P}_{Z_V|Y_V})$$
$$\triangleq \inf_{M \in \mathcal{M}(\mathsf{Z}|\mathsf{Z})} \|M \circ \mathbb{P}_{Z_U|Y_V} - \mathbb{P}_{Z_V|Y_V}\|_{\text{TV}}.$$

If $\delta(\mathbb{P}_{Z_U|Y_V} \to \mathbb{P}_{Z_V|Y_V}) = 0$ (no deficiency), we say that $\mathbb{P}_{Z_U|Y_V}$ is **sufficient** for $\mathbb{P}_{Z_V|Y_V}$. Defi-

---

[7] We refer to Sec. A.2 for more details.

ciency is a quantity in $[0, 1]$ which quantifies how informative $\mathbb{P}_{Z_U|Y_V}$ is about $\mathbb{P}_{Z_V|Y_V}$ (0 being perfectly informative).

Theorem 1 (0-deficiency).

$$\delta(\mathbb{P}_{Z_U|Y_V} \to \mathbb{P}_{Z_V|Y_V}) = 0 \Rightarrow V \widetilde{\subset} U.$$

The proof of this result is given in Sec. A.3. Restricting our definition of inclusion to 0-deficiency pairs of task is too restrictive and can rarely be achieved in practice, due to properties of the TV distance. We thus study the continuous spectrum of informativeness measured by the deficiency. We leverage additional results due to Le Cam (1964, 1996) that control the amount of information a task reveal about the other, function of the deficiency.

Theorem 2 (ε-deficiency (Le Cam, 1964)). *Let* $\varepsilon > 0$. *Then,* $\delta(\mathbb{P}_{Z_U|Y_V} \to \mathbb{P}_{Z_V|Y_V}) < \varepsilon$ *if and only if, for any bounded loss function ℓ, we have,*

$$\mathcal{R}_\ell(Y_V, Z_U) - \varepsilon \leqslant \mathcal{R}_\ell(Y_V, Z_V).$$

Where, $\mathcal{R}_\ell(Y_V, Z_U)$ *denotes the statistical risk of inferring Y_V from Z_U measured with the loss function ℓ*

Theorem 2 guarantees that the lower the deficiency the more V is included into U. The deficiency can be seen as the *missing* information.

3.3 Inclusion estimation

Deficiency proposed in Definition 3 is intractable in practice due to the complexity of TV distance (Bhattacharyya et al., 2023). However, Information Sufficiency (IS), originally introduced by Arimoto (1971) and more recently used in (Xu et al., 2020; Darrin et al., 2024b,a), can be an interesting proxy to estimate deficiency. IS of Z_U relatively to Z_V, denoted as $\mathcal{I}_S(Z_U \to Z_V)$, is a lower bound of the Mutual Information (MI) (Cover and Thomas, 2006, Section 2.3 p.20) $I(Z_U; Z_V)$ between embeddings, and is defined as,

$$\mathcal{I}_S(Z_U \to Z_V) \triangleq \hat{h}(Z_V) - \hat{h}(Z_V|Z_U), \quad (1)$$

where $\hat{h}$ denotes an estimation of the entropy and the conditional entropy[8]. Moreover, MI is a quantity that has links to Lenient inclusion of Definition 2 thanks to the following relations[9],

$$\begin{aligned} I(Z_U; Z_V) &\leqslant I(Z_U; Y_V), \\ I(Z_U; Z_V) &\leqslant I(Z_V; Y_U), \end{aligned} \quad (2)$$

---

[8] For more details about IS, we refer to Sec. A.4
[9] We provide a proof of these relations in Sec. C.3

where $I(Z_U; Y_V)$ can be seen as the "information" embeddings trained on U have on the response of V which is related to Definition 2. Then in terms of interpretation, the larger $\mathcal{I}_S(Z_U \rightarrow Z_V)$ is, the more Z_U is informative about Z_V thus the more information U carries about V, then the more included V is into U. Moreover, IS has been empirically tested as an interesting proxy to evaluate deficiency between embeddings (Darrin et al., 2024a). Lastly, we recall that IS is an unbounded positive value, ranging from 0 to $+\infty$ contrary to deficiency which lies in $[0, 1]$[10]. In the following, we will use $\mathcal{I}_S(Z_U \rightarrow Z_V)$ as a measure of how much task V is included in task U. The higher the value of $\mathcal{I}_S$ the more V will be considered as included into U. The main idea behind deficiency and its estimation through IS is to be able to simulate Z_V from Z_U. This has also been explored by Lange et al. (2021) in the context of task relationship, but with the use of the L_2 distances between samples of the measures with only a linear transformation allowed. Our setup is more general and allows for broader class of transformations with a reconstruction loss connected to the inclusion notion (c.f. Theorem 2).

4   Experimental setup

Two different settings are proposed here to quantify inclusion between tasks using the estimation of IS (c.f. Eq. 1) between embeddings. We will focus on inferring relations between pairs of tasks U and V using the following reasoning:

$$\mathcal{I}_S(Z_V \rightarrow Z_U) \leqslant \mathcal{I}_S(Z_U \rightarrow Z_V) \Rightarrow V \widetilde{\subset} U.$$

Synthetic experiment.   First, we consider a synthetic example to explore and validate the tool of IS as an inclusion metric. Task data is generated from a Hidden Markov Model (HMM) (Baum and Petrie, 1966) over a vocabulary of size 10, from which we sample sequences of up to 30 symbols. A standard 1-layer transformer-based language model is trained on these synthetic samples in a next-token prediction fashion. This *foundation model* is then fine-tuned on three simple classification tasks for which inclusion relationships are known. Given an input X made of a sequence $S = [s_1, \ldots, s_n]$ generated by the HMM, and two characters C_1 and C_2 drawn from the vocabulary, the three tasks are:

- First(S, C_1, C_2) (noted as F) should return $Y = 0$ if C_1 is the same as the first character of S ($C_1 = s_1$), 1 otherwise.

- Last(S, C_1, C_2) (noted as L) should return $Y = 0$ if C_2 is the same as the last character of S ($C_2 = s_n$), 1 otherwise.
- First_or_Last(S, C_1, C_2) (noted as F∨L) should return $Y = 0$ if C_1 is the first character of S and C_2 is the last ; else 1 if C_1 is the first, 2 if C_2 is the last ; 3 otherwise.

It is straightforward to see that the task F∨L includes the others while the converse is not true. For all tasks considered, representation X is only the output of the attention layer[11]. We generated 11 datasets based on various HMMs with a change in the emission of the Markov chain (leading to 33 models). All presented results are averaged over these different datasets. For further details about the setup of this experiment, we refer to Sec. D.1.

NLP Pipeline.   The second experimental setup serves as a proof of concept, demonstrating that our inclusion measure can be effectively applied to NLP tasks. We selected five classification tasks common in linguistic pipelines, syntactic parsing (SYN), semantic role labeling (SRL), named entity recognition (NER), and coreference resolution (COR), along with a text generation task: summarization (SUM). These tasks were chosen because their inclusion relationships can be linguistically defined through annotation schemes. We used the OntoNotes dataset (Pradhan and Xue, 2009), which provides multi-level linguistic annotations for news documents. SYN is based on the Penn Treebank (Marcus et al., 1993) annotation scheme, SRL on the Penn PropBank scheme, and NER uses eight OntoNotes categories for proper nouns (e.g., event, organization, person). COR annotations link coreferring noun phrases and, in some cases, verb phrases, making it a challenging task. For summarization, since OntoNotes lacks this annotation, we used GPT 3.5 to generate summaries for each document. From a linguistic perspective, annotation schemes exhibit direct inclusion relationships between syntactic (SYN) and semantic annotations (SRL). In contrast, the relationship between NER and both SYN and SRL is more interdependent: on one side, NER can leverage SYN and SRL to identify the left boundary of named entity spans; on the other side, NER can help defining noun phrases for SYN and SRL analyses. COR relies on cues from all other levels. Similarly, summarization requires syntactic and semantic simplification as well

---

[10]For more details, we refer to Sec. A.4.

[11]Last token of the sentence, since the model is causal.

Task	Description
SYN	List all `NP-SBJ` syntagms
SRL	List all predicates `PRED(ARG0,ARG1)`
NER	List all NEs for each `category`
COR	List all coreference chains
SUM	Summarize

Table 1: Generative adaptation of NLP pipeline tasks

	F	F∨L	L
F	0.736	0.236	0.130
F∨L	0.188	0.842	0.175
L	0.123	0.223	0.715

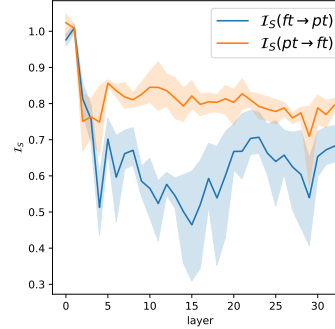Table 2: $\mathcal{I}_S(\text{row} \to \text{col})$ on pairs of synthetic tasks.

Figure 2: Layerwise information sufficiency between Mistral 7B base and that model model finetuned, averaged over the NLP pipeline tasks.

as structural compression, making knowledge of linked entities crucial. Our goal is to verify that IS can uncover these relationships. To standardize task processing, all tasks are reformulated as generative tasks. While summarization is inherently generative, linguistic tasks were adapted to extract relevant patterns under task-specific constraints. In order to simplify tasks and reduce sequence lengths, we restrict some tasks to generating a subset of the annotations (such as finding subjects for SYN), assuming that understanding the whole linguistic phenomenon is required to correctly generate such subset. Table 1 provides a detailed description of the reformulation; examples are listed in App. E. All tasks were applied at the document level. We used 1297, 98, and 97 documents as train, validation and test sets, from the broadcast news and newswire subsets of Ontonotes. We compute performance and IS across tasks on Mistral 7B (Jiang et al., 2023), and Llama 3 8B (Dubey et al., 2024) in their Instruct and Base versions. This choice is based on the relatively good results these models offer for their size, which allows repeated fine-tuning within a reasonable resource budget. These models were fine-tuned using Low Rank Adaptations (LoRA) (Hu et al., 2021) of rank 8, a regularization coefficient α of 16 and a learning rate of 4e-5 with a constant learning rate scheduler. This choice of adaptation is based on the low amount of data we have for training (≈ 1300) and the good properties LoRA offer in a case of low amount of data (Fu et al., 2023). Training is run for six epochs with a *best evaluation loss selection* strategy at the end of the training procedure. Fine-tunings were performed using the `transformers` (Wolf et al., 2020) and `peft` libraries from Huggingface.

5 Results

Synthetic experiment. In the context of the experiment on HMM data, we propose a deep analysis of the information sufficiency (IS). We provide on Table 2 IS results. The first thing that can be noticed is the presence of high mutual information

in the diagonal of the table, which is an important sanity check: the most informative task for one task is the task itself. The second thing we can notice is that the following property holds,

$$\mathcal{I}_S(\mathsf{F} \to \mathsf{L}) \leqslant \mathcal{I}_S(\mathsf{F} \vee \mathsf{L} \to \mathsf{L}),$$
$$\mathcal{I}_S(\mathsf{L} \to \mathsf{F}) \leqslant \mathcal{I}_S(\mathsf{F} \vee \mathsf{L} \to \mathsf{F}).$$

Which is in line with the relations between our tasks. Moreover, we have,

$$\mathcal{I}_S(\mathsf{F} \to \mathsf{F} \vee \mathsf{L}) \approx \mathcal{I}_S(\mathsf{L} \to \mathsf{F} \vee \mathsf{L}),$$

which is an interesting results considering that tasks F and L carry the same amount of information on F∨L. The converse observation can be made,

$$\mathcal{I}_S(\mathsf{F} \vee \mathsf{L} \to \mathsf{F}) \approx \mathcal{I}_S(\mathsf{F} \vee \mathsf{L} \to \mathsf{L}),$$

with a similar justification. These results suggest that IS provides an interesting proxy to estimate information about relations between tasks.

NLP Pipeline. Table 3 presents the results for each model across tasks, evaluated not on direct parser performance but on the generation process using RougeL scores (Lin, 2004) (as described in Table 1). As expected, coreference resolution is the most challenging task. For summarization, all models perform similarly in terms of RougeL. Instruct models improve results for Llama but not for Mistral, though these differences are minor given the small test corpus. First, we focus on the strategy

for determining IS between pairs of tasks, which can be computed from embeddings at any layer of the models. It has been empirically shown that different layers do not encode the same knowledge in LLMs: middle layers tend to focus more on the task the model is trained to solve, while last (deepest) layers focus more on the generation format of the task (Siddiqui et al., 2024; Zhang et al., 2024; Fischer et al., 2024). To support this fact, we ran a simulation for which we compared the hidden representations of the fine-tuned and the pre-trained model. More specifically, we compared $\mathcal{I}_S(Z_U \to Z)$ to $\mathcal{I}_S(Z \to Z_U)$, where Z refers to the embeddings of the pre-trained model. We report results in Figure 2 for the Mistral Base model (Figure 6 contains this figure for all models). Middle layers seem to stand out more from the pre-trained model, suggesting that they are the ones that seem to encode the task. This result is in line with recent work about redundancy between large language models' layers (Zhao et al., 2024b; Huang et al., 2024; Gromov et al., 2024; Men et al., 2024; González et al., 2025). Given this observation, in the following we will only look at average IS over layers 10-15, which seems to be layers for which the gap with the pre-trained model is the most significant. We propose in App. G an ablation study about the use of different layers, where we show the ineffectiveness of deep layers for task modelisation. In Figure 3, we plot a heat-map of the average IS across the four models[12]. First, summarization is probably the most elaborate task considering that IS from any linguistic task to summarization (column SUM) leads to the lowest values. The only informative task for summarization is itself. On the opposite we can see that SYN task seems to be included in every other task (high values in the associated columns). Moreover, we have,

$$\mathcal{I}_S(\text{SYN} \to \text{SRL}) \leqslant \mathcal{I}_S(\text{SRL} \to \text{SYN})$$
$$\mathcal{I}_S(\text{SRL} \to \text{NER}) \leqslant \mathcal{I}_S(\text{NER} \to \text{SRL}),$$

leading to the following lenient inclusion ranking, $\text{SYN} \,\widetilde{\subset}\, \text{SRL} \,\widetilde{\subset}\, \text{NER}$, which is completely in line with our premises about the linguistic pipeline. Finally, once again, COR task is the most challenging among linguistic tasks, as it seems to contain information about all linguistic tasks, while no other task seems to contain information about it (low values in the associated column). As for its comparison

---

	Base Llama Mistral		Instruct Llama Mistral	
SYN	97.6	97.5	97.6	97.3
SRL	81.5	80.5	82.0	81.8
NER	86.7	87.8	85.0	86.3
COR	53.9	61.2	53.7	61.7
SUM	48.8	49.6	49.6	48.5

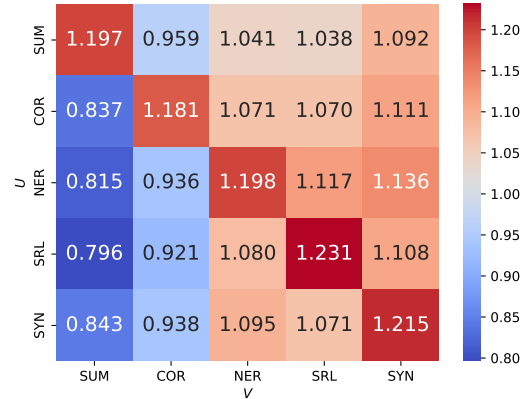Table 3: RougeL scores of LLMs on generative versions of NLP pipeline tasks. Rouge implementation comes from evaluate library.

Figure 3: Average of $\mathcal{I}_S(\text{row} \to \text{col})$ across models.

with summarization, we have,

$$\mathcal{I}_S(\text{COR} \to \text{SUM}) \leqslant \mathcal{I}_S(\text{SUM} \to \text{COR}),$$

which is once again in line with our initial thoughts on the linguistic pipeline. By putting all together these interpretations, clear hints seem to appear about the existence of the linguistic pipeline in the space of tasks (at least in the sense of the Ontonotes annotations mapped to generative tasks).

Predictive power. Interpretation of IS essentially relies on how much information one task contains about another *v.s.* how much information the others contain about the first one. Instead of having local interpretation of every combinations (which tends to increase rapidly) we can sum-up this idea by introducing a quantity we called the *predictive power* (PP). PP of a task U can be defined as

$$\text{PP}(U) \triangleq \sum_V \mathcal{I}_S(Z_U \to Z_V) - \mathcal{I}(Z_V \to Z_U).$$

Interpretation is direct: the higher PP is for a task, the more it contains information about the others while the others do not contain information about it (which is essentially interpretations we made earlier). Table 4 provides the increasing ranking of the different tasks in terms of predictive power for the

---

[12]Considering the multiple means, it is hard to provide proper confidence intervals.

	Avg.	Llama 3 Base Instruct		Mistral Base Instruct	
SUM	4.0	4	4	4	4
COR	3.0	3	3	3	3
NER	1.5	2	2	2	0
SRL	0.75	1	0	1	1
SYN	0.75	0	1	0	2

Table 4: PP ranking of the tasks for different models. The higher the ranking, the more informative about the others is the task. Avg. refers to the average ranking.

	F	F∨L	L
H	0.656	0.761	0.634

Table 5: Entropy of the hidden states (synthetic exp.)

different models (the higher the ranking is the more PP of the task is high). On average (across models), tasks order in terms of PP seems to respect our premises about the NLP pipeline. Additionally, we can see that base models respect the pipeline's order, while some noise is present on Instruct models. This can be contrasted with the fact that Instruct models are pre-trained to perform a wide range of tasks, which can disrupt task modeling and thus ranking (Mueller et al., 2024).

6 Discussion

Synthetic experiment. In Table 2 while some interesting interpretations can be done, problems remain as we have $\mathcal{I}_S(\mathsf{F∨L} \to .) \leqslant \mathcal{I}_S(. \to \mathsf{F∨L})$, which is not a desired property. However this property can largely be explained by the higher entropy of the hidden states of the task F∨L as shown in Table 5. This higher entropy will automatically give higher values of IS of the type $\mathcal{I}_S(. \to \mathsf{F∨L})$ which is basically illustrated on Table 2 (column F∨L)[13]. This higher entropy can be explained by the fact that F∨L is a 4-class classification problem while the others are 2-class.

Task inclusion. First, while we showed that we can mainly uncover the linguistic pipeline, we can observe similarity of our measures for tasks such as SRL and SYN (*i.e.* $\mathcal{I}_S(\mathsf{SYN} \to \mathsf{SRL}) \approx \mathcal{I}_S(\mathsf{SRL} \to \mathsf{SYN})$) thus questioning the significance. This result was expected in this case, as the SRL task considered here is closely related to the SYN task. To simplify the tasks and reduce sequence lengths, we restricted SRL and SYN to generating only a subset of annotations (ARG0

and ARG1 for SRL, and subjects and objects of verbs for SYN *c.f.* App. E). As a result, the two tasks are naturally closely aligned, a fact that is further highlighted by our metrics. Additionally, the setup we address in this work is rather simplistic. First, even among pipeline tasks, it is well known that there is no unidirectional relationship between tasks. For example, although semantic analysis is supposed to rely on syntactic analysis in linguistic pipelines, some syntactic constructs such as prepositional phrase attachment can only be disambiguated by semantic constraints (Brill and Resnik, 1994). Although statistical deficiency only allows for strict inclusion, the mutual information proxy represented by IS seems to be more in line with what we know from linguistics. Second, prompts can be written to create arbitrary combination of tasks, which would allow for very diverse, yet controlled, instances of inclusion. Should inclusion resulting from a logical operator be differentiated from inclusion resulting from sequential application of instructions? Finally, the end goal is probably to decompose tasks in a minimal non-overlapping set of skills, a notion which has been eluded in all benchmarking efforts so far[14].

7 Conclusion

This work aims at characterizing the space of NLP tasks through the notion of inclusion which we define as statistical deficiency. We propose IS as a tractable surrogate which allows comparing tasks from the embeddings of models trained on datasets annotated with those tasks. Experiments on synthetic and real NLP data suggest that this empirical notion of inclusion aligns with our preconception of task processing pipelines, potentially revealing which skills are required to perform some of the more elaborate tasks. Future work includes applying this framework to the selection of instruction tuning data by selecting most informative tasks/instructions within the data-mix to optimize dataset sizes without loss of performances. Another interesting line would be the conception of orthogonal evaluation benchmarks. We also plan on exploiting the task space structure for better handling task composition and generalization. A promising direction is to structure tasks as a Partial Ordering Set, that can be derived from a numerical application (Peleg, 1970) and which Shannon (1958) has applied to structure communication channels.

---

[13]For more details, we refer to Eq. 5

[14]In Sec. B.2 we formalize task decomposition.

8 Limitations

One of the main limitation of this work relies in the use of information sufficiency to estimate task inclusion. As we stated, IS is used as proxy to estimate deficiency. However computing IS, contrary to deficiency, does not account for response values Y_U and Y_V. These variables are part of the very essence of a task, as we can see from Definition 1. A more accurate way of estimating inclusion would be to directly estimate deficiency which will be addressed in future work. Under certain hypothesis, we can use other more tractable distances (Gibbs and Su, 2002), leading to other definitions of the deficiency. The other problem with this approach is that our inclusion estimate is empirical by nature, and relies on how representative the underlying corpus is of the general task distribution. This can be improved by collecting larger corpora but is fundamentally unbounded.

This leads to the second limitation of this work: estimating task inclusion from a single dataset, OntoNotes, in a single language, English. Hypothesis H2 requires that we use the same inputs for the compared tasks, and not many corpora offer this property. The tasks we cover are both limited in number and scope, only addressing a subset of the classic NLP pipeline, and are altered by framing them in a generative setting. In particular, considered NLP pipeline tasks involve only subsets of underlying linguistic structures, which weakens our task ordering claims. Besides, we only look at two average-sized LLMs given the combinatorics of model training/evaluation, and only use a single adaptation method, LoRA. Adaptation through fine-tuneing is one way of specializing models to perform a task, but we should explore zero-shot prompting and in-context learning as well (note that the proposed formalism is still valid in those cases). Those experiments should be seen as a proof of concept, not a complete proof, to confirm that the intuition of the linguistic pipeline can be rediscovered through the proposed metric. Future work is needed to extend the scope of results.

References

Alessandro Achille, Michael Lam, Rahul Tewari, Avinash Ravichandran, Subhransu Maji, Charless C Fowlkes, Stefano Soatto, and Pietro Perona. 2019. Task2vec: Task embedding for meta-learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6430–6439.

Alessandro Achille and Stefano Soatto. 2018. Emergence of Invariance and Disentanglement in Deep Representations. *Journal of Machine Learning Research*, 19(50):1–34.

Sydney N Afriat. 1957. Orthogonal and oblique projectors and the characteristics of pairs of vector spaces. In *Mathematical proceedings of the Cambridge philosophical society*, volume 53, pages 800–816. Cambridge University Press.

Eunice Akani, Benoit Favre, Frederic Bechet, and Romain Gemignani. 2023. Reducing named entity hallucination risk to ensure faithful summary generation. In *Proceedings of the 16th International Natural Language Generation Conference*, pages 437–442.

Suguru Arimoto. 1971. Information-theoretical considerations on estimation problems. *Information and Control*, 19(3):181–194.

Yajie Bao, Yang Li, Shao-Lun Huang, Lin Zhang, Lizhong Zheng, Amir Zamir, and Leonidas Guibas. 2019. An Information-Theoretic Approach to Transferability in Task Transfer Learning. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 2309–2313.

Leonard E Baum and Ted Petrie. 1966. Statistical inference for probabilistic functions of finite state markov chains. *The annals of mathematical statistics*, 37(6):1554–1563.

J. Baxter. 2000. A Model of Inductive Bias Learning. *Journal of Artificial Intelligence Research*, 12:149–198.

Sergey Berezin and Tatiana Batura. 2022. Named entity inclusion in abstractive text summarization. In *Third Workshop on Scholarly Document Processing*, page 158.

Arnab Bhattacharyya, Sutanu Gayen, Kuldeep S. Meel, Dimitrios Myrisiotis, A. Pavan, and N. V. Vinodchandran. 2023. On Approximating Total Variation Distance. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pages 3479–3487.

Ake Björck and Gene H Golub. 1973. Numerical methods for computing angles between linear subspaces. *Mathematics of computation*, 27(123):579–594.

David Blackwell. 1951. Comparison of Experiments. In Jerzy Neyman, editor, *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pages 93–102. University of California Press.

David Blackwell. 1953. Equivalent Comparisons of Experiments. *The Annals of Mathematical Statistics*, 24(2):265–272.

Malik Boudiaf, Jérôme Rony, Imtiaz Masud Ziko, Eric Granger, Marco Pedersoli, Pablo Piantanida, and Ismail Ben Ayed. 2021. A unifying mutual information view of metric learning: Cross-entropy vs. pairwise losses. *Preprint*, arXiv:2003.08983.

Eric Brill and Philip Resnik. 1994. A rule-based approach to prepositional phrase attachment disambiguation. In *COLING 1994 Volume 2: The 15th Int. Conference on Computational Linguistics*.

J. K. Brooks. 1971. The Lebesgue Decomposition Theorem for Measures. *The American Mathematical Monthly*, 78(6):660–662.

Rich Caruana. 1997. Multitask learning. *Machine learning*, 28:41–75.

Boli Chen, Yao Fu, Guangwei Xu, Pengjun Xie, Chuanqi Tan, Mosha Chen, and Liping Jing. 2021. Probing bert in hyperbolic spaces. In *International Conference on Learning Representations*.

T. M. Cover and J. A. Thomas. 2006. *Elements of Information Theory*, 2nd edition. Wiley, New York.

H. Crauel. 2002. *Random Probability Measures on Polish Spaces*. Taylor & Francis.

Maxime Darrin, Philippe Formont, Ismail Ben Ayed, Jackie CK Cheung, and Pablo Piantanida. 2024a. When is an Embedding Model More Promising than Another? *Preprint*, arXiv:2406.07640.

Maxime Darrin, Philippe Formont, Jackie Chi Kit Cheung, and Pablo Piantanida. 2024b. Cosmic: Mutual Information for Task-Agnostic Summarization Evaluation. *Preprint*, arXiv:2402.19457.

Thomas G Dietterich. 2000. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer.

David L. Donoho. 1988. One-Sided Inference about Functionals of a Density. *The Annals of Statistics*, 16(4):1390–1420.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Nadir Durrani, Hassan Sajjad, and Fahim Dalvi. 2021. How transfer learning impacts linguistic knowledge in deep NLP models? *Preprint*, arXiv:2105.15179.

Nadir Durrani, Hassan Sajjad, Fahim Dalvi, and Firoj Alam. 2022. On the Transformation of Latent Space in Fine-Tuned NLP Models. *Preprint*, arXiv:2210.12696.

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. 2022. Understanding dataset difficulty with v-usable information. In *International Conference on Machine Learning*, pages 5988–6008. PMLR.

Tim Fischer, Chris Biemann, et al. 2024. Large language models are overparameterized text encoders. *arXiv preprint arXiv:2410.14578*.

Zihao Fu, Haoran Yang, Anthony Man-Cho So, Wai Lam, Lidong Bing, and Nigel Collier. 2023. On the effectiveness of parameter-efficient fine-tuning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 12799–12807.

Lei Gao, Yue Niu, Tingting Tang, Salman Avestimehr, and Murali Annavaram. 2024. Ethos: Rectifying language models in orthogonal parameter space. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2054–2068, Mexico City, Mexico. Association for Computational Linguistics.

Alison L Gibbs and Francis Edward Su. 2002. On choosing and bounding probability metrics. *International statistical review*, 70(3):419–435.

Ramón Calvo González, Daniele Paliotta, Matteo Pagliardini, Martin Jaggi, and François Fleuret. 2025. Leveraging the true depth of llms. *arXiv preprint arXiv:2502.02790*.

Andrey Gromov, Kushal Tirumala, Hassan Shapourian, Paolo Glorioso, and Daniel A Roberts. 2024. The unreasonable ineffectiveness of the deeper layers. *arXiv preprint arXiv:2403.17887*.

Alain Guillaume and Bengio Yoshua. 2017. Understanding intermediate layers using linear classifier probes. In *ICLR (Workshop)*.

Steve Hanneke and Samory Kpotufe. 2024. A more unified theory of transfer learning. *arXiv preprint arXiv:2408.16189*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Hanjuan Huang, Hao-Jia Song, and Hsing-Kuo Pao. 2024. Large language model pruning. *arXiv preprint arXiv:2406.00030*.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing Models with Task Arithmetic. *Preprint*, arXiv:2212.04089.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Ruochen Jin, Bojian Hou, Jiancong Xiao, Weijie Su, and Li Shen. 2024. Fine-tuning linear layers only is a simple yet effective way for task arithmetic. *arXiv preprint arXiv:2407.07089*.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. 2023. Dataless Knowledge Fusion by Merging Weights of Language Models. *Preprint*, arXiv:2212.09849.

Olav Kallenberg. 2022. *Foundations of Modern Probability*. Springer Cham.

Robert W. Keener. 2010. *Theoretical Statistics*, 1st edition. Springer New York, NY, New York, NY.

Maurice G Kendall. 1938. A new measure of rank correlation. *Biometrika*, 30(1-2):81–93.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. UNIFIEDQA: Crossing format boundaries with a single QA system. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1896–1907, Online. Association for Computational Linguistics.

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. 2022. Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution. *Preprint*, arXiv:2202.10054.

Jenny Kunz and Marco Kuhlmann. 2020. Classifier probes may just learn from linear context features. In *Proc. of the 28th International Conference on Computational Linguistics*, pages 5136–5146.

Lukas Lange, Jannik Strötgen, Heike Adel, and Dietrich Klakow. 2021. To share or not to share: Predicting sets of sources for model transfer learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8744–8753, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

L. Le Cam. 1964. Sufficiency and Approximate Sufficiency. *The Annals of Mathematical Statistics*, 35(4):1419–1455.

L. Le Cam. 1996. Comparison of experiments—a short review. In *Institute of Mathematical Statistics Lecture Notes - Monograph Series*, pages 127–138. Institute of Mathematical Statistics, Hayward, CA.

Ziyan Li and Naoki Hiratani. 2025. Optimal task order for continual learning of multiple tasks. *arXiv preprint arXiv:2502.03350*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Xiaojun Xu, Yuguang Yao, Hang Li, Kush R Varshney, et al. 2024. Rethinking machine unlearning for large language models. *arXiv preprint arXiv:2402.08787*.

Saber Malekmohammadi and Golnoosh Farnadi. 2024. On the implicit relation between low-rank adaptation and differential privacy. In *NeurIPS 2024 Workshop on Mathematics of Modern Machine Learning*.

Mitch Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank. *Computational linguistics*, 19(2):313–330.

Andreas Maurer, Massimiliano Pontil, and Bernardino Romera-Paredes. 2016. The benefit of multitask representation learning. *Journal of Machine Learning Research*, 17(81):1–32.

Xin Men, Mingyu Xu, Qingyu Zhang, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. 2024. Shortgpt: Layers in large language models are more redundant than you expect. *arXiv preprint arXiv:2403.03853*.

Amil Merchant, Elahe Rahimtoroghi, Ellie Pavlick, and Ian Tenney. 2020. What Happens To BERT Embeddings During Fine-tuning? *Preprint*, arXiv:2004.14448.

Jianming Miao and Adi Ben-Israel. 1992. On principal angles between subspaces in rn. *Linear algebra and its applications*, 171:81–98.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. In *ACL*.

Marius Mosbach, Anna Khokhlova, Michael A. Hedderich, and Dietrich Klakow. 2020. On the Interplay Between Fine-tuning and Sentence-Level Probing for Linguistic Knowledge in Pre-Trained Transformers. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 68–82, Online. Association for Computational Linguistics.

David Mueller, Mark Dredze, and Nicholas Andrews. 2024. Multi-task transfer matters during instruction-tuning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14880–14891, Bangkok, Thailand. Association for Computational Linguistics.

Dmitry Nikolaev and Sebastian Padó. 2023. Investigating semantic subspaces of transformer sentence embeddings through linear structural probing. In *6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 142–154. Association for Computational Linguistics.

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. 2024. Task arithmetic in the tangent space: Improved editing of pre-trained models. *Advances in Neural Information Processing Systems*, 36.

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.

Bezalel Peleg. 1970. Utility functions for partially ordered topological spaces. *Econometrica: Journal of the Econometric Society*, pages 93–96.

Georg Pichler, Pierre Colombo, Malik Boudiaf, Günther Koliander, and Pablo Piantanida. 2022. A Differential Entropy Estimator for Training Neural Networks. *Preprint*, arXiv:2202.06618.

Georg Pichler, Pablo Piantanida, and Günther Koliander. 2021. On the Estimation of Information Measures of Continuous Distributions. *Preprint*, arXiv:2002.02851.

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. 2020. Information-theoretic probing for linguistic structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4609–4622, Online. Association for Computational Linguistics.

Sameer S Pradhan and Nianwen Xue. 2009. Ontonotes: the 90% solution. In *Proceedings of Human Language Technologies: NAACL 2009*, pages 11–12.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR.

Maxim Raginsky. 2011. Shannon meets Blackwell and Le Cam: Channels, codes, and statistical experiments. In *2011 IEEE International Symposium on Information Theory Proceedings*, pages 1220–1224, St. Petersburg, Russia. IEEE.

Mark D Reid and Robert C Williamson. 2011. Information, divergence and risk for binary experiments. *Journal of Machine Learning Research*, 12(3).

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2021. A primer in bertology: What we know about how bert works. *Transactions of the Association for Computational Linguistics*, 8:842–866.

Jürgen Schmidhuber. 1987. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. Ph.D. thesis, Technische Universität München.

Ozan Sener and Vladlen Koltun. 2018. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31.

Claude E Shannon. 1958. A note on a partial ordering for communication channels. *Information and control*, 1(4):390–397.

Changjian Shui, Mahdieh Abbasi, Louis-Émile Robitaille, Boyu Wang, and Christian Gagné. 2019. A principled approach for learning task similarity in multitask learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 3446–3452.

Shoaib Ahmed Siddiqui, Xin Dong, Greg Heinrich, Thomas Breuel, Jan Kautz, David Krueger, and Pavlo Molchanov. 2024. A deeper look at depth pruning of llms. In *ICML 2024 Workshop on Theoretical Foundations of Foundation Models*.

Yang Tan, Yang Li, and Shao-Lun Huang. 2021. OTCE: A Transferability Metric for Cross-Domain Cross-Task Representations. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15774–15783, Nashville, TN, USA. IEEE.

Zhixu Tao, Ian Mason, Sanjeev Kulkarni, and Xavier Boix. 2024. Task Arithmetic Through The Lens Of One-Shot Federated Learning. *Preprint*, arXiv:2411.18607.

Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. 2020. Measuring Robustness to Natural Distribution Shifts in Image Classification. In *Advances in Neural Information Processing Systems*, volume 33, pages 18583–18599. Curran Associates, Inc.

Naftali Tishby, Fernando C. Pereira, and William Bialek. 2000. The information bottleneck method. *Preprint*, arXiv:physics/0004057.

Lisa Torrey and Jude Shavlik. 2010. Transfer learning. In *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, pages 242–264. IGI global.

Alan M. Turing. 1950. *Computing Machinery and Intelligence*, pages 23–65. Springer Netherlands, Dordrecht.

Tu Vu, Tong Wang, Tsendsuren Munkhdalai, Alessandro Sordoni, Adam Trischler, Andrew Mattarella-Micke, Subhransu Maji, and Mohit Iyyer. 2020. Exploring and predicting transferability across NLP tasks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7882–7926, Online. Association for Computational Linguistics.

Andreas Waldis, Yotam Perlitz, Leshem Choshen, Yufang Hou, and Iryna Gurevych. 2024. Holmes: A Benchmark to Assess the Linguistic Competence of Language Models. *Preprint*, arXiv:2404.18923.

Jonas Wallat, Fabian Beringer, Abhijit Anand, and Avishek Anand. 2023. Probing bert for ranking abilities. In *European Conference on Information Retrieval*, pages 255–273. Springer.

Xinyi Wang, Antonis Antoniades, Yanai Elazar, Alfonso Amayuelas, Alon Albalak, Kexun Zhang, and William Yang Wang. 2024. Generalization v.s. memorization: Tracing language models' capabilities back to pretraining data. In *ICML 2024 Workshop on Foundation Models in the Wild*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Terry Winograd. 1987. *Thinking machines: Can there be? Are we?* Department of Computer Science, Stanford University.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. 2022. Robust fine-tuning of zero-shot models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7949–7961, New Orleans, LA, USA. IEEE.

Sen Wu, Hongyang R. Zhang, and Christopher Ré. 2020. Understanding and Improving Information Transfer in Multi-Task Learning. *Preprint*, arXiv:2005.00944.

Yilun Xu, Shengjia Zhao, Jiaming Song, Russell Stewart, and Stefano Ermon. 2020. A theory of usable information under computational constraints. In *International Conference on Learning Representations*.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A. Raffel, and Mohit Bansal. 2023. TIES-Merging: Resolving Interference When Merging Models. *Advances in Neural Information Processing Systems*, 36:7093–7115.

Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. 2023. AdaMerging: Adaptive Model Merging for Multi-Task Learning. *Preprint*, arXiv:2310.02575.

Jiasheng Ye, Peiju Liu, Tianxiang Sun, Yunhua Zhou, Jun Zhan, and Xipeng Qiu. 2024. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. *Preprint*, arXiv:2403.16952.

Qinyuan Ye. 2024. Cross-task generalization abilities of large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*, pages 255–262, Mexico City, Mexico.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. 2020. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33:5824–5836.

Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. 2018. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3712–3722.

Siqi Zeng, Yifei He, Weiqiu You, Yifan Hao, Yao-Hung Hubert Tsai, Makoto Yamada, and Han Zhao. 2025. Efficient model editing with task vector bases: A theoretical framework and scalable approach. *arXiv preprint arXiv:2502.01015*.

Jinghan Zhang, Junteng Liu, Junxian He, et al. 2023a. Composing parameter-efficient modules with arithmetic operation. *Advances in Neural Information Processing Systems*, 36:12589–12610.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. 2023b. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Yang Zhang, Yanfei Dong, and Kenji Kawaguchi. 2024. Investigating layer importance in large language models. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 469–479, Miami, Florida, US. Association for Computational Linguistics.

Yu Zhang and Qiang Yang. 2021. A survey on multi-task learning. *IEEE transactions on knowledge and data engineering*, 34(12):5586–5609.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2024a. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2):1–38.

Zheng Zhao, Yftah Ziser, and Shay B Cohen. 2024b. Layer by layer: Uncovering where multi-task learning happens in instruction-tuned large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15195–15214, Miami, Florida, USA. Association for Computational Linguistics.

Pan Zhou, Yingtian Zou, Xiao-Tong Yuan, Jiashi Feng, Caiming Xiong, and Steven Hoi. 2021. Task similarity aware meta learning: Theory-inspired improvement on maml. In *Uncertainty in artificial intelligence*, pages 23–33. PMLR.

Yuyan Zhou, Liang Song, Bingning Wang, and Weipeng Chen. 2024. MetaGPT: Merging Large Language Models Using Model Exclusive Task Arithmetic. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1711–1724, Miami, Florida, USA. Association for Computational Linguistics.

A   Details about the formalism

In this section we give additional details about Sec. 3.

A.1   Measures and Kernels

In this study, we assume that all considered spaces are standard Borel (Crauel, 2002). Each such space X is equipped with its Borel σ-algebra $\mathcal{B}(\mathsf{X})$. Having this regularity about the topology on our spaces is necessary to have equivalence between conditional probabilities and Markov transition kernels (Kallenberg, 2022, Theorem 8.5 p.168), assuring the existence of conditional probabilities. Moreover, for every random variable $X \in \mathsf{X}$, we denote by $\mathbb{P}_X$ the push-forward measure induced by X on X. Thus considering two random variables $X \in \mathsf{X}$ and $Y \in \mathsf{Y}$, we have $\mathbb{P}_{Y|X} \in \mathcal{M}(\mathsf{Y}|\mathsf{X})$. We additionally recall here some basic properties on measures and Markov kernels, that are used during this study.

Definition 4 (Total variation distance)**.** Given two probability measures P and Q on $(\mathsf{X}, \mathcal{B}(\mathsf{X}))$,

$$\|P - Q\|_{\mathrm{TV}} \triangleq \sup_{b \in \mathcal{B}(\mathsf{X})} |P(b) - Q(b)|.$$

Total variation distance between probability measures induces a distance on Markov Kernels' space, which can be defined as following, for two kernels M and K in $\mathcal{M}(\mathsf{Y}|\mathsf{X})$.

$$\|K - M\|_{\mathrm{TV}} = \sup_{x \in \mathsf{X}} \|K(.|x) - M(.|x)\|_{\mathrm{TV}}.$$

Deficiency in Definition 3, uses a composition operation between Markov kernels, that is defined as following,

Definition 5 (Markov composition operation)**.** Let $K \in \mathcal{M}(\mathsf{Z}|\mathsf{Y})$ and $M \in \mathcal{M}(\mathsf{Y}|\mathsf{X})$,

$$(K \circ M)(b|z) = \int_{\mathsf{Y}} K(b|y) M(dy|z)$$

This composition operation must be viewed as a generalization of the law of total probability.

A.2   Embeddings as Kernels

In this study, $\mathbb{P}_{Z_U|Y_U} \in \mathcal{M}(\mathsf{Z}|\mathsf{Y})$ is used to describe embeddings, especially in Definition 3. This kernel describes the shape of Z_U given particular values of $y \sim \mathbb{P}_{Y_U}$. An interesting case to understand this kernel is classification tasks, for which Y is a finite discrete space. It is a well known fact that for classification tasks, $\mathbb{P}_{Z_U|Y_U}$ can be described as a cluster-type model. In that case $\mathbb{P}_{Z_U|Y_U}(.|y)$ corresponds to the cluster of Z_U associated to $Y_U = y$. When looking at $\mathbb{P}_{Z_U|Y_V}$ we seek to understand if Z_U has also a clustering-type structure which makes sens for Y_V and then if we can infer values of Y_V from Z_U, which is echoing the inclusion of $\mathbb{P}_{XY_V}$ in $\mathbb{P}_{XY_U}$.

A.3   Proof of Theorem 1

We provide here the proof of Theorem 1. This result, states that a 0-deficiency can be seen as an inclusion between two tasks. As a recall the relaxed version of the inclusion of V into U states that solving task U is informative to solve task V. Theorem 1 states that this useful information obtained from one task for the other, is contained in the embeddings.

The proof of Theorem 1 requires another assumption about the fine-tuning operation. More particularly if Z_U are the embeddings of a text X, provided by a model trained on the task $\mathbb{P}_{XY_U}$, then Z_U can achieve the best error rate (Bayes risk) on task $\mathbb{P}_{XY_U}$, which is equivalent as,

$$\begin{aligned}
\mathcal{R}_\ell(Y_U, Z_U) &\triangleq \inf_{d \in \mathcal{M}(\mathsf{Y}_U|\mathsf{Z}_U)} \mathbb{E}_{y \sim Y_U} \mathbb{E}_{\hat{y} \sim d \circ \mathbb{P}_{Z_U|Y_U}(.|y)} \ell(y, \hat{y}) \\
&= \inf_{d \in \mathcal{M}(\mathsf{Y}_U|\mathsf{X})} \mathbb{E}_{y \sim Y_U} \mathbb{E}_{\hat{y} \sim d \circ \mathbb{P}_{X|Y_U}(.|y)} \ell(y, \hat{y}),
\end{aligned} \tag{3}$$

for any bounded loss function l. Second infimum is achieved for the *posterior*, $\mathbb{P}_{Y_U|X}$, meaning that there exists $T_U \in \mathcal{M}(\mathsf{Y}|\mathsf{Z})$ such that,

$$T_U \circ \mathbb{P}_{Z_U|X} = \mathbb{P}_{Y_U|X}. \tag{4}$$

Proof. If $\delta(\mathbb{P}_{Z_U|Y_V} \to \mathbb{P}_{Z_V|Y_V}) = 0$, then there exists some $K \in \mathcal{M}(\mathsf{Z}|\mathsf{Z})$, such that $K \circ \mathbb{P}_{Z_U|Y_V} = \mathbb{P}_{Z_V|Y_V}$. The statistical risk of the task $\mathbb{P}_{XY_V}$ from Z_V being given by,

$$\mathcal{R}_\ell(Y_V, Z_V) = \inf_{d \in \mathcal{M}(\mathsf{Y}|\mathsf{Z})} \mathbb{E}_{y \sim Y} \mathbb{E}_{\hat{y} \sim d \circ \mathbb{P}_{Z_V|Y_V}(.|y)} \ell(y, \hat{y}),$$

By data-processing (Reid and Williamson, 2011; Raginsky, 2011), we thus have,

$$\mathcal{R}_\ell(Y_V, Z_U) \leqslant \mathcal{R}_\ell(Y_V, Z_V) \quad \forall \ell \text{ s.t. } \|l\|_\infty \leqslant 1.$$

However, Z_V is supposed to reach Bayes risk, implying thus that Z_U also reaches Bayes risk for Y_V. Consequently there exists $T \in \mathcal{M}(\mathsf{Y}|\mathsf{Z})$ such that,

$$T \circ \mathbb{P}_{Z_U|X} = \mathbb{P}_{Y_V|X}.$$

Thus, by solving task U, *i.e.* by inferring the posterior $\mathbb{P}_{Y_U|X}$ through embeddings Z_U, we are also able to produce the posterior $\mathbb{P}_{Y_V|X}$ of the task V, which concludes the proof. $\qquad\square$

A.4 Information Sufficiency definition

As stated in the core of the study, while deficiency is the true measure of the inclusion, it is too complex to estimate forcing us to use information sufficiency. Information sufficiency is a lower bound of the mutual information between embeddings Z_U and Z_V. We have,

$$I(Z_U; Z_V) = H(Z_U) - H(Z_U|Z_V)$$
$$= H(Z_V) - H(Z_V|Z_U).$$

While we do not have access to the true distributions, calculus of the above entropy is impossible. Information sufficiency estimates the distributions of the embeddings, with a maximum $\log$-likelihood estimation, by making the assumption that distributions lie in a parametric family. Information sufficiency has thus the following expression,

$$\mathcal{I}_S(Z_U \to Z_V) \triangleq \hat{H}_{\mathcal{P}_\theta(\mathsf{Z})}(Z_V) - \hat{H}_{\mathcal{M}_\Theta(\mathsf{Z}|\mathsf{Z})}(Z_V|Z_U)$$
$$\mathcal{I}_S(Z_V \to Z_U) \triangleq \hat{H}_{\mathcal{P}_\theta(\mathsf{Z})}(Z_U) - \hat{H}_{\mathcal{M}_\Theta(\mathsf{Z}|\mathsf{Z})}(Z_U|Z_V), \tag{5}$$

where $\hat{H}_{\mathcal{P}_\theta(\mathsf{Z})}$ is the estimation of the entropy on the class $\mathcal{P}_\theta(\mathsf{Z})$. In our case we use KNIFE estimator (Pichler et al., 2022) to compute this quantity. This estimator chooses Gaussian Mixtures for the parametric family. This choice can be justified by the fact that the set of Gaussian Mixtures is dense for the weak topology (convergence in probability) within the set of Lebesgue-continuous probability measures, assuring we can approximate every continuous probability measure with a Gaussian Mixture. Moreover, Gaussian mixtures have interesting properties in terms of smoothness. It has been shown in (Donoho, 1988) and confirmed in (Pichler et al., 2021) that estimating mutual information for distribution with no particular smoothness assumptions (particularly distributions respecting the *dense graph condition*) is impossible from a finite sample of the distribution (whatever the size), justifying once again the choice of Gaussian Mixture here.

One of the main justification of the fact that information sufficiency is an interesting proxy for estimating deficiency, lies in the fact that the term $\hat{H}_{\mathcal{M}_\Theta(\mathsf{Z}|\mathsf{Z})}(Z_V|Z_U)$ is an estimation of the amount of information, one embedding is carrying about the other and gives an interesting idea how, we can re-construct one embedding from another.

B A measure theoretic view of tasks

In this work, we defined tasks as probability measures $\mathbb{P}_{XY}$ on a product space $(\mathsf{X} \times \mathsf{Y})$. In this section we propose interpretations of classical measure theory results to show links with the inclusion proposed in Definition 2.

B.1 About domain shift (H2 assumption)

The objective of this study is to compare tasks in terms of the needed skills to solve them[15]. To do so, we made the hypothesis H2 about the marginal distributions on the texts of our tasks. First, this assumption is also made in connected works (Bao et al., 2019) in particular set-ups this assumptions can be relaxed (Tan et al., 2021). We further justify here this hypothesis, and show that by fixing the marginal $\mathbb{P}_X$, when comparing two tasks (not through a fine-tuned model), we compare the skills needed to solve them. This is done through disintegration Theorem (Kallenberg, 2022, Theorem 8.5 p.168). Given a task $\mathbb{P}_{XY}$, there exists a unique ($\mathbb{P}_X$ a-s) $M \in \mathcal{M}(Y|X)$, such that for every $E \in \mathcal{B}(X \times Y)$,

$$
\begin{aligned}
\mathbb{P}_{XY}(E) &= (\mathbb{P}_X \otimes M)(E) \\
&= \int_X \int_Y \mathbb{1}_E(x,y)(\mathbb{P}_X \otimes M)(dx,dy) \\
&= \int_Y \int_X \mathbb{1}_E(x,y)\mathbb{P}_X(dx)M(dy|x)
\end{aligned}
$$

Where $\otimes$, only denotes the product operator between measures defines on different measure spaces. In the following we will denote such kernel M as $\mathbb{P}_{Y|X}$. This decomposition assures that whatever the task we are considering, we can disentangle the domain and the skills needed to solve the task,

$$
\mathbb{P}_{XY} = \underbrace{\mathbb{P}_X}_{\text{Domain}} \otimes \underbrace{\mathbb{P}_{Y|X}}_{\text{Skills}}
$$

Considering this disentanglement, we'll assume that comparison of tasks will not lead to interpretation about the domain.

Example 1 (Difference). *Given two tasks $\mathbb{P}_{XY_U}$ and $\mathbb{P}_{XY_V}$, and $E \in (\mathcal{B}(X \times Y))$,*

$$
\mathbb{P}_{XY_U}(E) - \mathbb{P}_{XY_V}(E) = \int_X \int_Y \mathbb{1}_E(x,y)\mathbb{P}_X(dx)(\mathbb{P}_{Y_U|X}(dy|x) - \mathbb{P}_{Y_V|X}(dy|x))
$$

In the case of same skills, this difference is equal to zero, and the value of the difference is strongly linked to differences in the skills required for tasks

B.2 Measure theoretic view of inclusion

In this Section we explore the possibilities to understand the behavior of a task, w.r.t another set of tasks, in a measure theoretic point of view *i.e.* by directly comparing tasks without using any proxy. A classical result in measure theory is the Lebesgue decomposition Theorem (Brooks, 1971), which states that one task can be decomposed interestingly by using another task, echoing a notion of shared information (and thus inclusion) between two tasks. The Theorem is the following: given two tasks, $\mathbb{P}_{XY_U}$ and $\mathbb{P}_{XY_V}$, we have,

$$
\mathbb{P}_{XY_U} = \mu_V + \rho \quad \text{such that} \quad \mu_V << \mathbb{P}_{XY_V} \text{ and } \rho \perp \mathbb{P}_{XY_V}. \tag{6}
$$

In the case of such decomposition, μ_V can be seen of the "information" $\mathbb{P}_{XY_U}$ encodes on $\mathbb{P}_{XY_V}$. By doing such interpretation, we implicitly consider that domination between measures is a sort of inclusion metric. We show below that this interpretation can be true.

Given two tasks $\mathbb{P}_{XY_U}$ and $\mathbb{P}_{XY_V}$, such that $\mathbb{P}_{XY_V} << \mathbb{P}_{XY_U}$ then for every $E \in \mathcal{B}(X \times Y)$, there exists a unique (up to a $\mathbb{P}_{XY_U}$ null measure space), positive real valued function f (which is called the density), such that,

$$
\begin{aligned}
\mathbb{P}_{XY_V}(E) &= \int_X \int_Y \mathbb{1}_E(x,y)f(x,y)\mathbb{P}_{XY_U}(dx,dy) \\
&= \int_X \int_Y \mathbb{1}_E(x,y)f(x,y)\mathbb{P}_X(dx)\mathbb{P}_{Y_U|X}(dy|x).
\end{aligned} \tag{7}
$$

---

[15]The skills here refer to the conditional measure $\mathbb{P}_{Y|X}$

Having a domination relation, allows us to express the $\mathbb{P}_{XY_V}$-measure of every event E, with the use of $\mathbb{P}_{Y_U|X}$, which is related to Definition 2, *i.e.* having $\mathbb{P}_{Y_U|X}$ is informative about $\mathbb{P}_{Y_V|X}$. In the situations where $\mathbb{P}_{XY_V} << \mathbb{P}_{XY_U}$ we can state the $\mathbb{P}_{XY_V} \subset \mathbb{P}_{XY_U}$.

Remark 2 (Domination in practice). In practice, we do not have access to theoretical probability measures, but we have access to samples from this measure. We therefore do not work on the theoretical measures but on the quantized version of these measures that we denote by $\tilde{\mathbb{P}}_{XY_U}$ and $\tilde{\mathbb{P}}_{XY_V}$. In that case, we define S_U and S_V the samples from respectively $\mathbb{P}_{XY_U}$ and $\mathbb{P}_{XY_V}$. In this case a sufficient condition conditions to have $\tilde{\mathbb{P}}_{XY_V} << \tilde{\mathbb{P}}_{XY_U}$ is that, $S_V \subset S_U$.

Thus domination is interestingly related to the notion of task inclusion. One can thus ask how to find a similar set up of the one proposed with the information sufficiency in this study, but with using domination. We propose here several definitions to be able to express one task with respect to other ones,

Definition 6 (Independent tasks). Let $\{\mathbb{P}_{XY_i}, \ i \in \{1, \dots, N\}\}$ a finite set of tasks. We say that this set is independent iff,

$$\mathbb{P}_{XY_i} \perp \mathbb{P}_{XY_j} \quad \forall i \neq j$$

Proposition 1. *Let $\mathbb{P}_{XY}$ a task, and $\{\mathbb{P}_{XY_i}, \ i \in \{1, \dots, N\}\}$ be an independent set of tasks, then we have the following decomposition exists,*

$$\mathbb{P}_{XY} = \sum_{i=1}^N \mu_i + \rho \quad \text{such that} \quad \mu_i << \mathbb{P}_{XY_i}, \quad \text{and} \quad \rho \perp \mathbb{P}_{XY_i} \quad \forall i$$

Moreover this decomposition is uniquely verified by the different tasks. ρ is the remaining of the decomposition, i.e. the task we can't explain from our independent set of tasks.

Example 2 (Interpretation of the remaining ρ). *Given two independent tasks $\mathbb{P}_{XY_U}$ and $\mathbb{P}_{XY_V}$, we construct a new task $\mathbb{P}_{XY}$, such that $Y = f(Y_U, Y_V)$, where f is a (possibly randomized) transformation. Accordingly to Proposition 1, we have,*

$$\mathbb{P}_{XY} = \mu_U + \mu_V + \rho.$$

In this case ρ can be viewed as the contribution of the application f.

However, in Proposition 1, the term ρ remains hard to interpret. It would be interesting to have $\rho = 0$, meaning that we have a set of task that is **complete** for our target task $\mathbb{P}_{XY}$

Definition 7 (Complete set of tasks). We say that an independent set of tasks $\{\mathbb{P}_{XY_i}, \ i \in \{1, \dots, N\}\}$ is complete for a task $\mathbb{P}_{XY}$, when,

$$\mathbb{P}_{XY} = \sum_{i=1}^N \mu_i \quad \text{such that} \quad \mu_i << \mathbb{P}_{XY_i}.$$

Remark 3. Let $\mathbb{P}_{XY}$ a task and $T \triangleq \{\mathbb{P}_{XY_i}, \ i \in \{1, \dots, N\}\}$ an independent set of tasks.

$$T \text{ is complete for } \mathbb{P}_{XY} \Rightarrow \mathbb{P}_{XY} << \sum_i \mathbb{P}_{XY_i}$$

Proof. Let T be complete for $\mathbb{P}_{XY}$. Thus,

$$\mathbb{P}_{XY} = \sum_i \mu_i.$$

Let $E \in \mathcal{B}(\mathsf{X} \times \mathsf{Y})$, such that $\sum_i \mathbb{P}_{XY_i}(E) = 0$. Because we use non-signed measures, this implies $\mathbb{P}_{XY_i}(E) = 0 \ \forall i$, and consequently $\mu_i(E) = 0 \ \forall i$, by construction of the μ_i. We thus have the following property,

$$\forall E \in \mathcal{B}(\mathsf{X} \times \mathsf{Y}), \quad \sum_i \mathbb{P}_{XY_i}(E) = 0 \Rightarrow \mathbb{P}_{XY}(E) = 0$$

Which concludes the proof. □

Having a complete set of task for a target task $\mathbb{P}_{XY}$ means that given this set of task, we can fully determine this task. Having a complete independent set of task can be seen as having a **basis of the task space**. From the basis some notions can be derived such as the dimension (cardinal of the basis) of the space. One can also define the complexity of a task, as the cardinality of the smallest set of independent task which is complete for the task (of course this set should not contain the task itself).

C Information theoretic view of fine-tuning

Information theory (Cover and Thomas, 2006) provides powerful tools to understand the dynamic of model trainings. We explore here this framework to make connections between learning theory and tasks inclusion. We first recall that given two random variables Y and Z, $I(Y;Z)$ refers to the mutual information between these variables and is defined as,

$$
\begin{aligned}
I(Y;Z) &= H(Y) - H(Y|Z) \\
&= H(Z) - H(Z|Y).
\end{aligned}
$$

C.1 Cross entropy decomposition

If we refer to Figure 1, and we suppose that our models are trained using cross entropy loss function $\mathcal{H}$ (which is our case during all this study), then it has been shown (Boudiaf et al., 2021) that this loss function can be decomposed in two terms,

$$
\mathcal{H}(Y_U, \hat{Y}_U) = H(Y_U|Z_U) + D_{\text{KL}}(Y_U || \hat{Y}_U | Z_U).
$$

Since $H(Y_U)$ is a constant during training, minimizing the cross entropy loss is equivalent to minimizing the following loss function:

$$
\mathcal{L}(Y_U, \hat{Y}_U) = \underbrace{-I(Y_U; Z_U)}_{\text{Task Info}} + \underbrace{D_{\text{KL}}(Y_U || \hat{Y}_U | Z_U)}_{\text{Task alignment}}. \tag{8}
$$

In this decomposition of the cross-entropy function the first term in mutual information refers to the information the model captures about the task *i.e.* the information the model captures to infer the response. The second term refers to the alignment of the estimation with the true labels. When training a model by minimizing the cross entropy we seek to create embeddings Z_U of text X that capture as much information about Y_U as possible and that are aligned with the distribution $\mathbb{P}_{Y_U}$. Thus when a model is trained on a task $\mathbb{P}_{XY_U}$ it might not be able to perform the task $\mathbb{P}_{XY_V}$ due to a misalignment (second term) and thus it is not respecting the first task inclusion property as we defined it. However, the quantity $I(Y_V; Z_U)$ can be great suggesting that the model captures information about the task.

Example 3 (Fully informative but miss-aligned). *Let $\mathbb{P}_{XY_U}$ a task, and $Z_U \in \mathbb{R}^d$ the representation of the text X given by a model trained using $\mathcal{H}$. We additionally suppose that $\hat{Y}_U = f(Z_U)$, where f is optimized w.r.t. Y_U. Let ϕ be a permutation in $\{1, \ldots, d\}$. Because ϕ is an invertible mapping,*

$$
I(Y_U; Z_U) = I(Y_U; \phi(Z_U)).
$$

However, we have $\hat{Y}_U^\phi = f(\phi(Z_U))$ and thus we have,

$$
D_{KL}(Y_U || \hat{Y}_U | Z_U) \leqslant D_{KL}(Y_U || \hat{Y}_U^\phi | \phi(Z_U))
$$

Thus based on Z_U we can construct $\phi(Z_U)$, which is as informative as Z_U for the task $\mathbb{P}_{XY_U}$, while it will perform worse due to a misalignment.

Remark 4 (Probing interpretation). From Eq. 8, we can make a remark about current probing methods, which mainly consist of training a linear probe on top of the representation Z_U without modifying them. Thus, when we probe a representation Z_U on the task $\mathbb{P}_{XY_V}$, using cross entropy, we only optimize the following quantity

$$
D_{\text{KL}}(Y_V || \hat{Y}_V | Z_U),
$$

which is not directly linked to the information Z_U captures about the task $\mathbb{P}_{XY_V}$. Probing simply answers the question of whether Z_U can be aligned with Y_U using a linear extractor, which can be restrictive. Current interpretation of probing would be correct if one uses sufficiently powerful probe as stated in (Pimentel et al., 2020).

C.2 Sufficiency of the models

As we can see in Eq. 8, when fine-tuning a model on a task $\mathbb{P}_{XY_U}$ with cross entropy loss (which is our case in this study), the model constructs embeddings Z_U such that $I(Y_U; Z_U)$ is maximized (Boudiaf et al., 2021). However, by *data processing inequality* (Cover and Thomas, 2006, Chapter 2, Section 8, p.34) we have the following inequality,

$$I(Y_U; Z_U) \leqslant I(Y_U; X).$$

Thus maximizing the mutual information $I(Y_U; Z_U)$ is equivalent in approximate $I(Y_U; X)$. Thus the fine-tuning process tends to produce embeddings Z_U such that,

$$I(Y_U; Z_U) \approx I(Y_U; X).$$

This is equivalent in saying that fine-tuning a model on a task $\mathbb{P}_{XY_U}$ produces embeddings that are sufficient statistics (in Fisher's sense) of X for Y_U (Achille and Soatto, 2018). This justifies the choice of using the embeddings as a proxy to represent the tasks, and proceed to the inclusion calculation.

C.3 Mutual Information: an interesting proxy

We give additional results that justify the use of the mutual information as a proxy to estimate the task inclusion. Eq. 8 gives an interesting decomposition of the cross-entropy loss function giving the interpretation that the inclusion measure of a task through the study of a model, can be viewed as the estimation of $I(Y_V; Z_U)$ (resp. $I(Y_U; Z_V)$). However, Figure 1 can be reduced to the following Markov chains $Z_U \leftrightarrow Y_V \leftrightarrow Z_V$, $Z_V \leftrightarrow Y_U \leftrightarrow Z_U$. These relations are true in the case of single reference tasks (*i.e.* the case where for an input data $x \sim \mathbb{P}_X$ there is only one single possible answer $y \sim \mathbb{P}_Y$), which is our case for any all linguistic tasks defined. Thus by *data processing inequality*, we have the following relations,

$$I(Z_U; Z_V) \leqslant I(Y_V; Z_U),$$
$$I(Z_U; Z_V) \leqslant I(Y_U; Z_V).$$

Thus the information sufficiency as a lower bound of the true mutual information, is a lower bound of $I(Y_V; Z_U)$ and $I(Y_U; Z_V)$ which can be viewed as inclusion estimation for models fine-tuned using cross-entropy.

D Details about the trainings

D.1 Synthetic experiment

As stated in this study we provided an experiment on a synthetic formal language generated from Hidden Markov Models (HMM). We produced 11 datasets following different HMM distributions. To generate different datasets, we fixed the underlying automaton of the HMM and we only changed the emission probabilities. The used HMM are simple ones, described on Figure 4. Presented results in this study are averaged on the different datasets we used. We provide on Table 6 the parameters we used for pre-training and fine-tuning on HMM generated data. First of all to check that the pre-training of our transformer based language models has worked we compared the forward likelihood of the HMM to the estimated likelihood of the transformer model. Results are provided on Figure 5. We can clearly see that the pre-trained transformer model approximate better the forward distribution compared to a non-trained model.
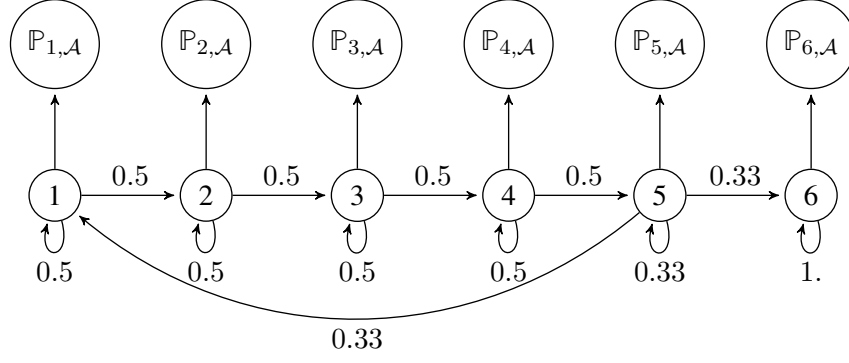
Figure 4: Illustration of the used markov chain for data generation. The quantity $\mathbb{P}_{i,\mathcal{A}}$ refer to the emission probabilities of each states.

	Pre-training	Fine-tuning
input size	50	50
hidden size	100	100
# layer	1	1
# heads	1	1
lr	2e-03	2e-04
lr-scheduler	cosine	cosine
# epochs	100	100
# batch	200	200

Table 6: Description of the hyper-parameters for the training of transformer based models on described HMM.

	F1 Micro	F1 Macro	Acc
F	0.81 (0.16)	0.78 (0.20)	0.81 (0.16)
F∨L	0.61 (0.20)	0.54 (0.27)	0.61 (0.20)
L	0.85 (0.11)	0.82 (0.15)	0.85 (0.11)

Table 7: Mean-accuracy (over the different datasets) of the different classification tasks.
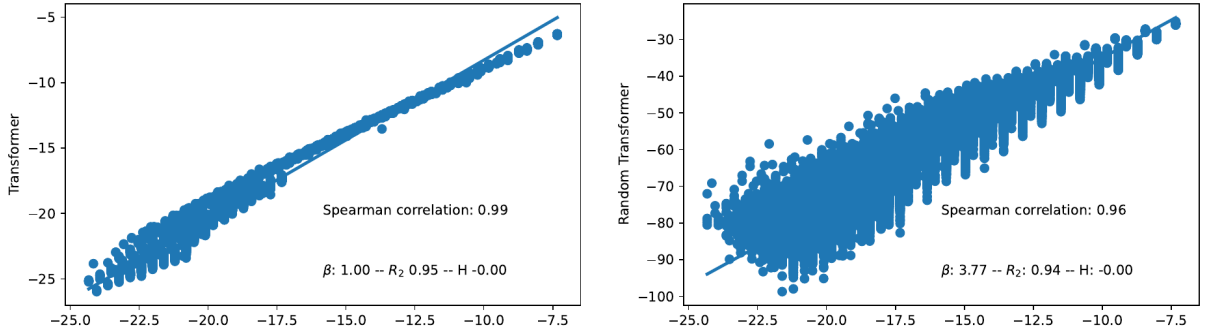
Figure 5: HMM forward likelihood *v.s.* empirical likelihood of the transformer based model

Parameter	Value
# Marg. Epochs	100
# Cond. Epochs	100
Marg. Lr	0.0001
Cond. Lr	0.001
# FF-layers	2
# Marg. Modes (Marginal Gaussian Mixture)	8
# Cond. Modes (Conditional Gaussian Mixture)	8
Covariance type	Diagonal

Table 8: KNIFE hyper-parameters. Selection of hyper-parameters is based on the extensive study provided in (Darrin et al., 2024a)

D.2 Information Sufficiency estimation

We provide on Table 8 hyper-parameters of the KNIFE estimator. The only change was made for the production of graphics on Figure 6 for which we limited our training on 20 epochs for sake of computational cost efficiency, since the goal of this simulation was mainly to understand the role of each layer.

E Details about the data

We provide below, an example of the data we used along with the different annotation schemes.

Input. Voters in strife torn Colombia go to the polls today in local elections in the midst of a wave of violence directed by armed groups of the left and the right against many of the candidates. VOA 's Bill Rodgers has a report from our South American bureau. Some 23 million Colombians are registered to vote Sunday to elect governors, mayors and other local officials in the South American nation, but the election is taking place in the midst of a rising wave of assassinations, kidnappings and threats against the candidates by leftist guerillas and right wing paramilitary groups. Twenty mayoral candidates have been killed and more than 200 kidnapped in recent months. Election authorities say 0 another 200 politicians withdrew their candidacies after being threatened. At the same time, these armed groups are backing reported to have fielded a number of stealth candidates in an effort to expand its control of Colombian territory. Despite the violence, the government of President Andres Pastrana has refused to suspend Sunday's elections vowing 0 they will take place throughout the country and in a democratic atmosphere. Bill Rodgers, VOA News, South America Bureau.

SUM. Colombia faces violence from armed groups targeting candidates in local elections; government vows to continue voting as planned.

COR. "strife - torn Colombia" refers also to "the South American nation" and "the government of President Andres Pastrana" and "the country". "local elections" refers also to "the election". "armed groups of the left and the right" refers also to "leftist guerillas and right - wing paramilitary groups" and "these armed groups". "VOA 's" refers also to "our" and "VOA News". "Bill Rodgers" refers also to "VOA 's Bill Rodgers". "South America Bureau" refers also to "VOA 's South American bureau". "Sunday" refers also to "Sunday 's". "the violence" refers also to "a rising wave of assassinations , kidnappings and threats against the candidates by these armed groups".

NER. This text contains the following entity name : "Colombia" and the following list of dates : "today, Sunday, recent months" and the following list of organisation names : "VOA, VOA News" and the following list of person names: "Bill Rodgers, Andres Pastrana" and the following list of person types: "South American, Colombians, Colombian" and the following list of numbers: "23 million, Twenty, more than 200, 200" and the following facility name: "South America Bureau".

SEM. have.03("VOA's","a report"); vote.01("Some 23 million","to"); say.01("Election authorities","0"); withdraw.01("another 200","their candidacies"); field.01("these armed groups","a number"); refuse.01("the government","to"); suspend.01("the government","Sunday 's"); vow.01("the government","0").

SYN. Voters; VOA 's Bill Rodgers; Some 23 million Colombians; the election; Twenty mayoral candidates; more than 200; Election authorities; another 200 politicians; these armed groups; the government; they.

F Additional results

In this section, we provide additional results that were not presented as main results, but which can help on the comprehension of the presented results.

F.1 Layer selection

To select layers and ease the estimation of IS, we provided a simulation on which we compared fine-tuned models to pre-trained models in terms of information sufficiency. IS is firstly used in this study as an inclusion metric. However, its first formulation is a lower bound of the mutual information. Thus when estimating the IS from a fine-tuned model to a pre-trained model we estimate the amount of information the model as lost or gain after the fine-tuning operation, compared to the pre-trained model. Figure 6 provides results of such comparison for all models present in this study. The first thing we can notice, is that the fine-tuning operation makes the model loose information about the pre-trained model ($\mathcal{I}_S(ft \rightarrow pt) \leqslant \mathcal{I}_S(pt \rightarrow ft)$), as if the fine-tuning operation consisted in applying a mask on the pre-trained model. The second observation and it is the one that is used in this study, is that layers where this gap is the biggest is between 10 and 15 suggesting that it is in-between these layers that the fine-tuned model has lost most information about the pre-trained model, suggesting that the task is mostly encoded here, which is the reason why we used these layers.

F.2 Information sufficiency matrices

Additionally to Figure 3, we provide onFigure 7 information sufficiency matrices for all the models. As a complement of Table 4, we provide on Figure 8 details of the PP ranking of the different tasks and for the different models. A positive PP for a task means that the task contains the others while the others do not contain this task. We can see that the only tasks with positive PP are summarization and COREF, pointing once again the difficulty of these two tasks and the need for these tasks of various linguistic skills to be performed correctly, which is once again in line with premises about the NLP pipeline.

F.3 Correlation with the naive approach

One of the most intuitive way to measure tasks' inclusion is to train a model on a task U and evaluate its performances on a task V. Considering we are using generative language models in this study, this can easily be done by a simple change of prompt. Thus we can measure the performance of a model trained on U on the task V, using classical metric such as BERTScore (Zhang* et al., 2020) or ROUGE (Lin, 2004). This new evaluation of the inclusion gives new matrices such as the one presented on Figure 7, except that these new matrices present the metric performances of a cross task evaluation. We present on Table 9 correlations in terms of Kendall-τ (Kendall, 1938) coefficient between IS matrices and cross task performances for ROUGE and BERTScore. Low correlation results suggest that there is no clear correlations between these two approaches. The main explanation being the alignment problem between tasks, considering that the output format for each task is different.

G Layer ablation study

In our experiments, we proposed a method to select the most relevant layers on which to calculate IS. We propose here an ablation study on the layer selection. We propose here several variations of Table 4, based on different layer selections, to understand which layer is interesting to discover the NLP pipeline. Our first analysis, is based on layers from 10 to 15 based on Figure 7. Our main argument being that deepest layers (after 15) are not relevant to understand the behavior of the task with respect to the model.
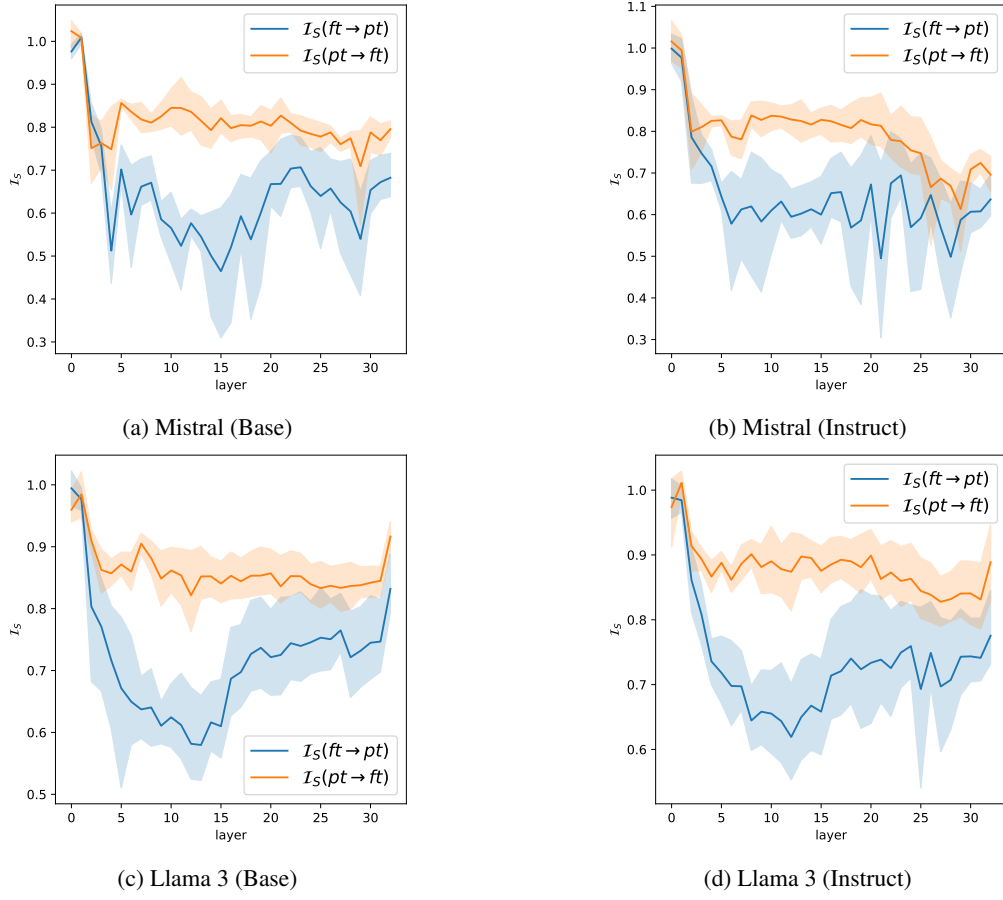
(a) Mistral (Base)　　　　　　　　　(b) Mistral (Instruct)

(c) Llama 3 (Base)　　　　　　　　　(d) Llama 3 (Instruct)

Figure 6: Information sufficiency comparison between pre-trained models and each corresponding fine-tuned model.

	BERT	**ROUGE**
Llama 3 base	0.02	0.43
Llama 3 instruct	-0.03	0.37
Mistral base	0.23	0.43
Mistral instruct	0.05	0.29

Table 9: Kendall-τ between information sufficiency and naive cross evaluation set-up.

(a) Mistral Base

(b) Mistral Instruct

(c) Llama 3 Base
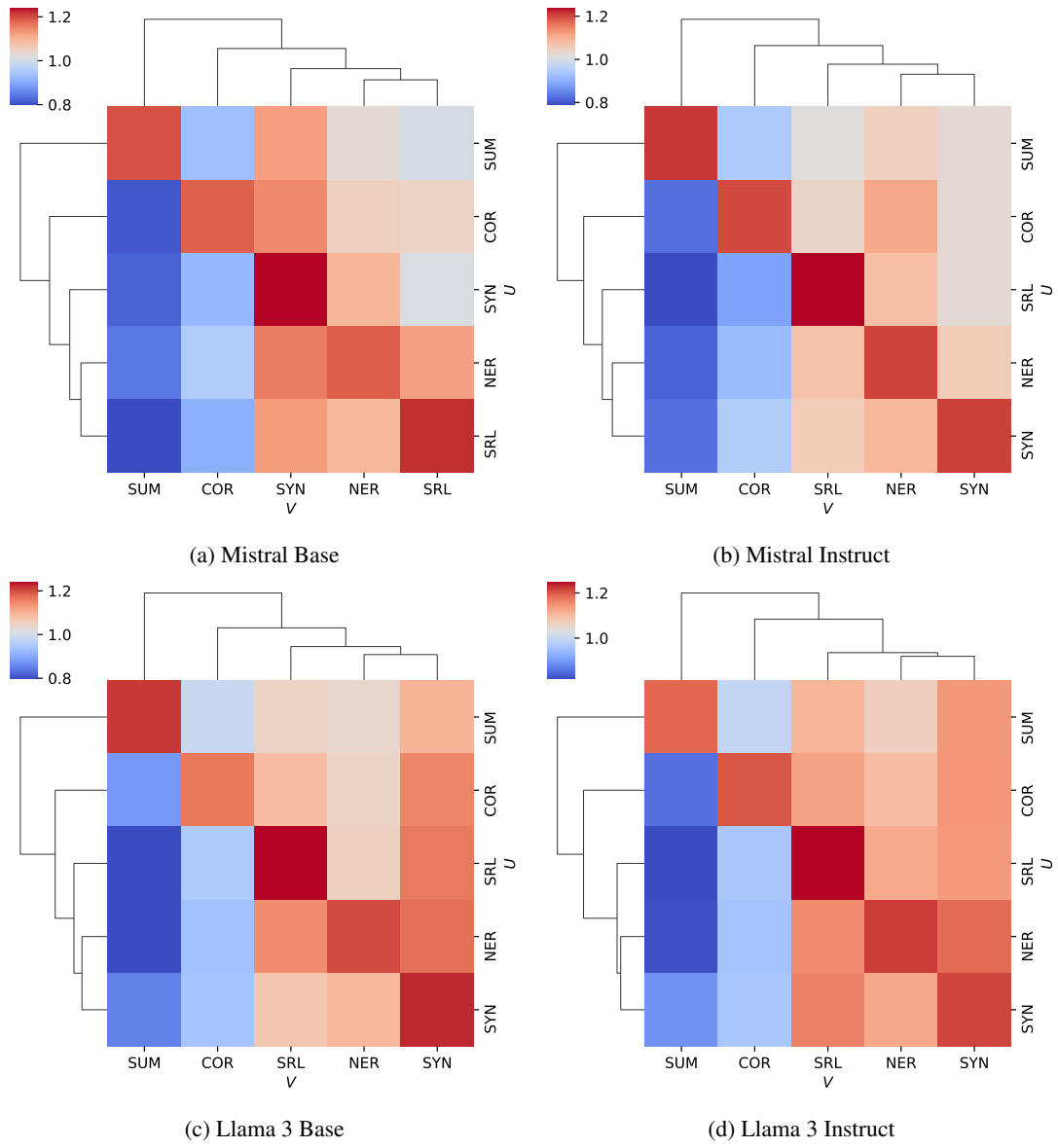
(d) Llama 3 Instruct

Figure 7: Information sufficiency averaged between layers 10 and 15.
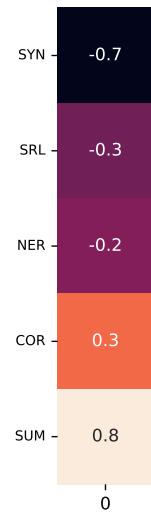
(a) Mistral Base

(b) Mistral Instruct

(c) Llama 3 Base

(d) Llama 3 Instruct

Figure 8: Information sufficiency averaged between layers 10 and 15.

	Avg.	Llama 3 Base	Llama 3 Instruct	Mistral Base	Mistral Instruct
SUM	4.0	4	4	4	4
COR	3.0	3	3	3	3
SRL	1.25	2	0	2	1
NER	1.0	1	2	1	0
SYN	0.75	0	1	0	2

Table 10: Ranking on all layers

	Avg.	Llama 3 Base	Llama 3 Instruct	Mistral Base	Mistral Instruct
SUM	4.0	4	4	4	4
COR	3.0	3	3	3	3
SRL	1.25	2	0	2	1
NER	1.0	1	2	1	0
SYN	0.75	0	1	0	2

Table 11: Ranking on layer 10 to 33

	Avg.	Llama 3 Base	Llama 3 Instruct	Mistral Base	Mistral Instruct
SUM	4.0	4	4	4	4
COR	3.0	3	3	3	3
NER	1.75	2	2	2	1
SRL	0.75	1	1	1	0
SYN	0.5	0	0	0	2

Table 12: Ranking on layer 1 to 20

First, on Table 10 we propose the same analysis, based on all the layers. We can observe that in that case, the NLP pipeline is not fully respected with a change of position of NER and SRL. In order to better understand the layers that can actually interfere with pipeline discovery, we are carrying out two new rankings. The first is based on layers 10 to 33 (preservation of deep layers) and is reported on Table 4 on which we can once again see the perturbation between SRL and NER. Then we perform the same ranking with layers between 1 and 20 (focus on lower layers) which we report on Table 12. On this last ranking we can see that the same pipeline as the one presented in the main study is respected. We can see that deepest layers can effectively introduce noise to discover relationship between tasks, which is once again in line with known works. In order to better visualize this behavior, we report on Figure 9, the evolution of the values of the IS for our different task combinations. We observe interesting behaviors mainly between layers 10 and 15, with a spike in IS values, suggesting that IS finds relationships between tasks at this level, while in the deeper layers, values are lower, which can once again be explained by differences in output formats. We can observe high values of IS in the very early layers, which can largely be explained that lower layers will mostly remain unchanged during fine-tuning due to gradient vanishing problems.

Generally speaking, we can see that changes in the layers only affect the position of two tasks (NER and SRL) while the other ones remain essentially untouched. What's more, we can see from this analysis that in these different set-ups, the Base models deliver the same information overall, while the instructed models differ in the given rankings, as highlighted in the body of the study.

H Task vector approach

Task vectors (Ilharco et al., 2023) are objects of growing interest in the community. They were firstly defined in the case of transfer learning, where a pre-trained model is fine-tuned on different downstream tasks. In that case, the task vector was defined as follows:

Definition 8. Let $W_0 \in \mathbb{R}^{m \times n}$, be the weights of a pre-trained model[16], and let W_U be the weights of the

---

[16]The weights of a model can always be represented as a matrix or a vector. For some connection with current results, we
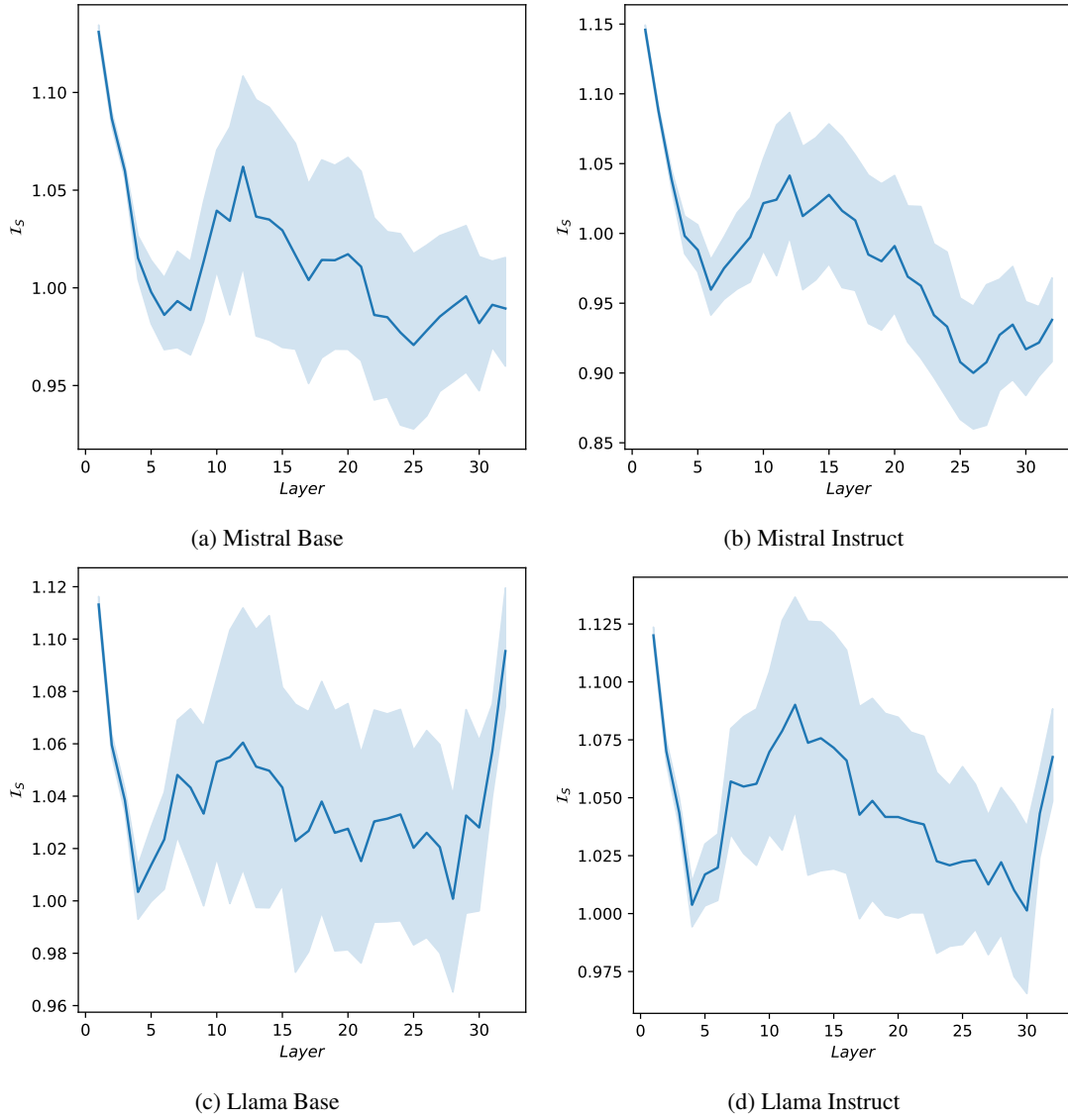
(a) Mistral Base

(b) Mistral Instruct

(c) Llama Base

(d) Llama Instruct

Figure 9: Information sufficiency evolution across layers.

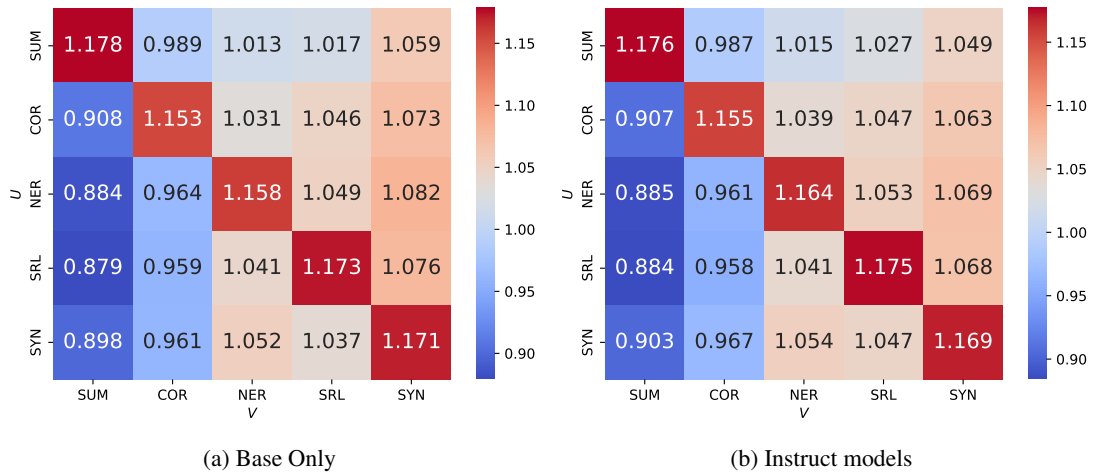(a) Base Only

(b) Instruct models

Figure 10: IS heat, for layers ranging from 1 to 20 with an average over only Base Models (Figure 10a), or on only Instruct models (Figure 10b).

same model, but after fine-tuning it on a task $U \equiv \mathbb{P}_{XY_U}$. The task vector of task U is given by:

$$\tau_U = W_U - W_0. \tag{9}$$

These objects were used to combine properties of a model through arithmetic operations (Ortiz-Jimenez et al., 2024; Jin et al., 2024; Zhang et al., 2023a), or to remove certain components such as toxicity, personal information or bias from a language model (Gao et al., 2024; Zhang et al., 2023a; Liu et al., 2024). In this study we used Low Rank Adaptation (LoRA (Hu et al., 2021)) for our fine-tunings on the different tasks. In this special case we have,

$$W_U = W_0 + B_U A_U \quad \text{with} \quad B_U \in \mathbb{R}^{m \times r}, \ A_U \in \mathbb{R}^{r \times n},$$

where $r \in \mathbb{N}$ is the chosen rank (which is 8 in this study). Given this definition of LoRA, we have,

$$\tau_U = B_U A_U.$$

In this special case, comparing task vectors, is equivalent as comparing different product $B_U A_U$. We propose here several distances in order to compare task vectors.

Cosine distance. One of the main distances to compare semantically two vectorial objects is the cosine similarity. We propose here to simply define this distance as following,

$$d_{\cos}(\tau_U, \tau_V) \triangleq 1 - \cos\left(\text{flatten}(B_U A_U), \text{flatten}(B_V A_V)\right).$$

L_2 **distance.** A standard way to compare vectorial representations is through euclidean distances, which we define as following,

$$d_{L_2}(\tau_U, \tau_V) \triangleq \|\text{flatten}(B_U A_U) - \text{flatten}(B_V A_V)\|_2.$$

Due to the dimension of task vectors euclidean distances are really restrictive.

Grassmann distance. Another way to compare task vectors is to see them as vector spaces. In fact when using LoRA, task vectors are defined through matrices which define vector spaces (column space). Grassmann distance is a mathematically well defined distance between vector spaces. It is based on the notion of principal angles between vector spaces (Afriat, 1957; Miao and Ben-Israel, 1992). If W_U and W_U are two matrices of rank r whose columns are orthonormal[17], then we can consider σ, the set of eigenvalues of $W_U^T W_V$. A theoretical result given by Björck and Golub (1973) is that these eigenvalues are the cosines of the principal angles between the image spaces of W_1 and W_2 I.e. if we set θ to be the set of principal angles between $\text{Im}(W_U)$ and $\text{Im}(W_V)$, then $\cos(\theta) = \sigma$. Based on this information, we then have :

$$d_G(W_U, W_V) = \sqrt{\sum_{i=1}^{r} (\theta_i)^2} \leqslant \sqrt{r}\frac{\pi}{2}. \tag{10}$$

An additional remark on this Grassmann distance is that if we consider $(W_U, W_V) \in \mathbb{R}^{d \times d}$, two matrices of maximum rank d, then $\text{Im}(W_U) = \text{Im}(W_V) = \mathbb{R}^d$ implying that $d_G(W_1, W_2) = 0$. This distance is therefore only of interest for matrices that are not of maximal rank, making it particularly interesting in the context of LoRA. Based on this, in our context, the Grassmann distance between task vectors will only be,

$$d_G(\tau_U, \tau_V) \triangleq d_G(B_U A_U, B_V A_V).$$

Grassmann distance has already been used in (Hu et al., 2021, Appendix G.) in the context of measuring the covering between different LoRA adaptations. In our context, the covering is between different task vectors, and thus by extrapolation, between different tasks. An interesting property about Grassmann distance, is the following,

---

chose the matrix representation.

[17]if they are not, a simple singular value decomposition of the matrices can allow us to do so

Proposition 2 (Grassmann distance is A-invariant.). *In the case of LoRA of rank r, if $rank(A_U) = rank(A_V) = r$, then we have,*

$$d_G(B_U A_U, B_V A_V) = d_G(B_U, B_V).$$

Proof. The proof is direct by using the rank Theorem. □

Remark 5. Proposition 2 requires the following condition,

$$\text{rank}(A_U) = \text{rank}(A_V) = r.$$

However, as stated in (Malekmohammadi and Farnadi, 2024), when using LoRA, A matrices remain essentially unchanged, and because they are initialized randomly by using a Gaussian distributions, the probability of having full rank A matrices is 1. Moreover in our experiments, we checked this property empirically, allowing thus to use Proposition 2.

Remark 6. Every results presented here is to be interpreted as distances (the higher the less similar).

H.1 Link with Information sufficiency.

Language models we are using are essentially continuous and differentiable applications with respect to its parameters. Thus a small change in the parameters of the models will induce a small change in the outputs of the models (this is direct application of the definition of continuity). Implying thus that closeness in the task vectors will induce little variation in the activation space, the inverse being not necessary true. This implies that a small distance between task vectors, will most likely provide high information sufficiency. However a high information sufficiency can be discovered between task vectors that are far from each other.

H.2 Result analysis

In the context of our models, we decided to apply LoRA on query and value projections on the different layers of the models. Thus for every tasks we defined and for every models, we have,

$$\tau_t^Q \triangleq (B_t^{Q,l}, A_t^{Q,l}) \quad \text{For the Query projection bloc at layer } l$$
$$\tau_t^V \triangleq (B_t^{V,l}, A_t^{V,l}) \quad \text{For the Value projection bloc at layer } l$$

Since in the literature, it is a well known fact that Query and Value projection encode different information, we decided to separate the analysis between Queries and Values, by looking at the average distance across layer for each module, *i.e.*

$$\text{d}\left(\tau_U^Q, \tau_V^Q\right) \triangleq \frac{1}{L} \sum_{l=1}^{L} \text{d}\left(B_U^{Q,l} A_U^{Q,l}, B_V^{Q,l} A_V^{Q,l}\right) \quad \text{Query distance}$$

$$\text{d}\left(\tau_U^V, \tau_V^V\right) \triangleq \frac{1}{L} \sum_{l=1}^{L} \text{d}\left(B_U^{V,l} A_U^{V,l}, B_V^{V,l} A_V^{V,l}\right) \quad \text{Value distance}$$

(11)

Figure 11 and Figure 12 provide distance results for the Mistral model (respectively Base and Instruct). Figure 13 and Figure 14 provide same results for the Llama 3 model. First we can see that results seem similar between task vectors on the Values and the Queries. Then, in terms of Grassmann and Cosine distances, we have a closeness between SYN and SRL, which is in line with our initial results on IS. In the same way as for IS, we observe the specific character of the summarization task, with a great distance from the other tasks present. The L_2 distance does not give interesting association between tasks (with respect to the underlying pipeline hypothesis), for the different studied models. This can easily be explained by the restrictive nature of the L_2 distance in such a high-dimensional space.
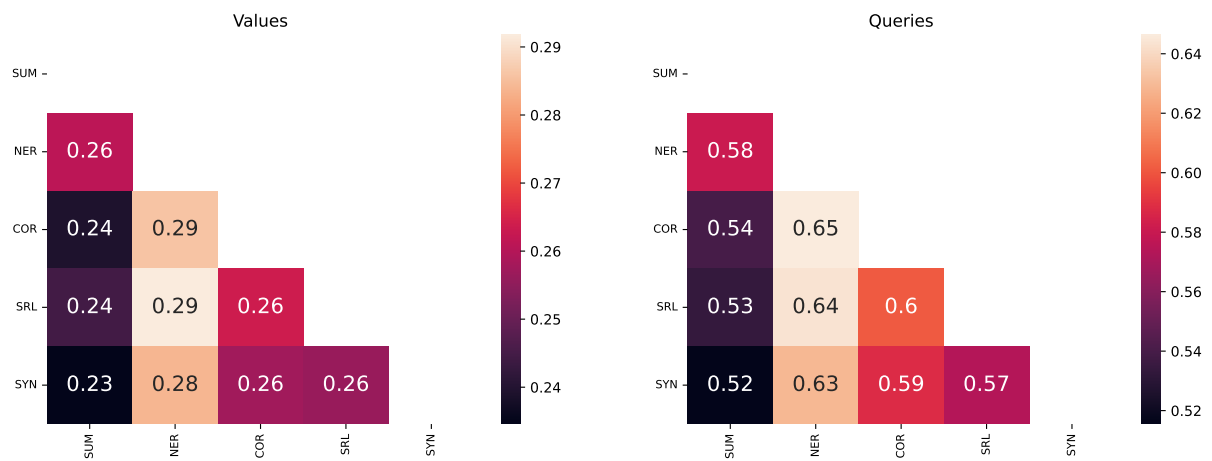
H.3 Limitations

One of the main limitation of this approach is its symmetric character. When assessing distances between task vectors, we rely on a symmetric approach which is not leading to interpretation of some partial ordering between tasks, which is our main goal in this study.
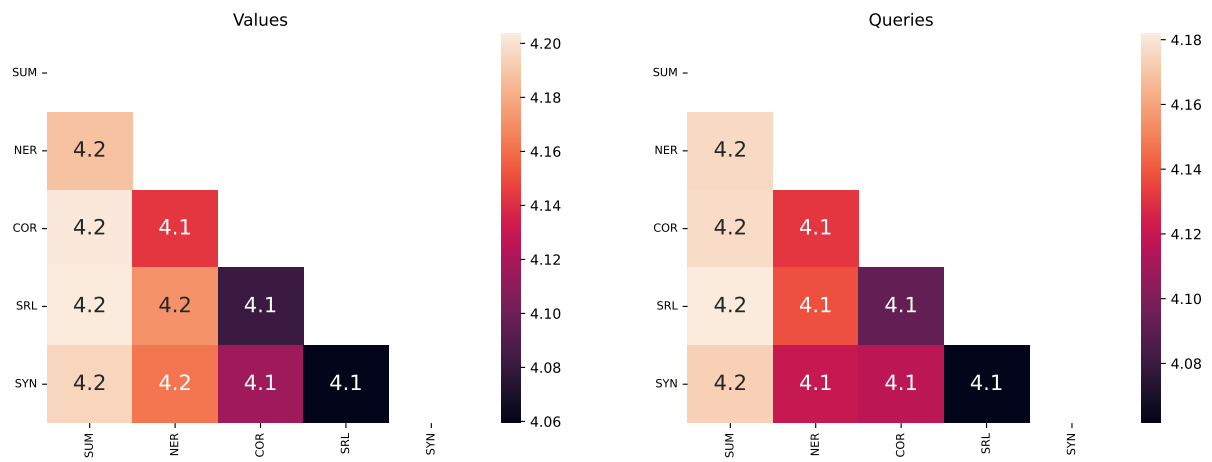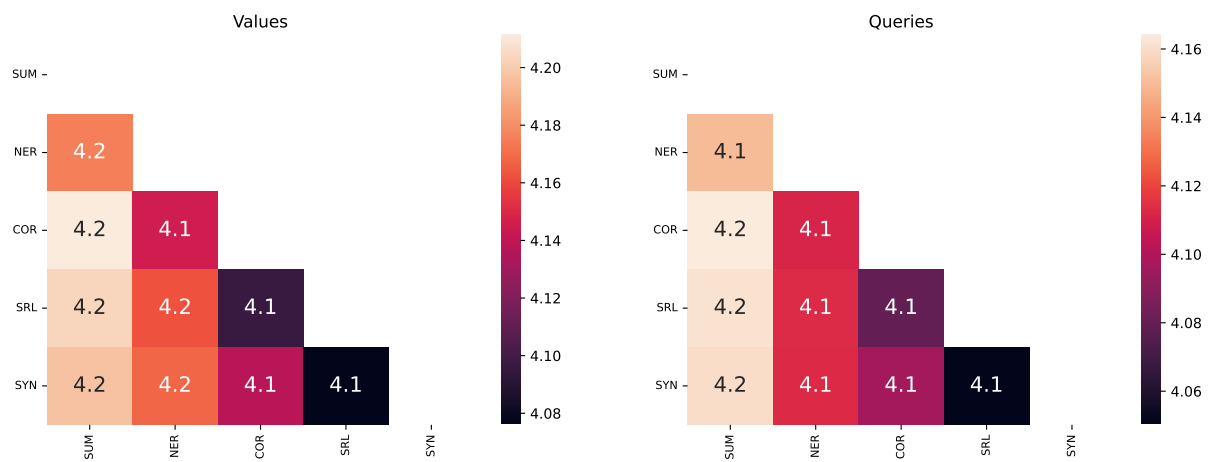
(a) Grassmann Mistral Base
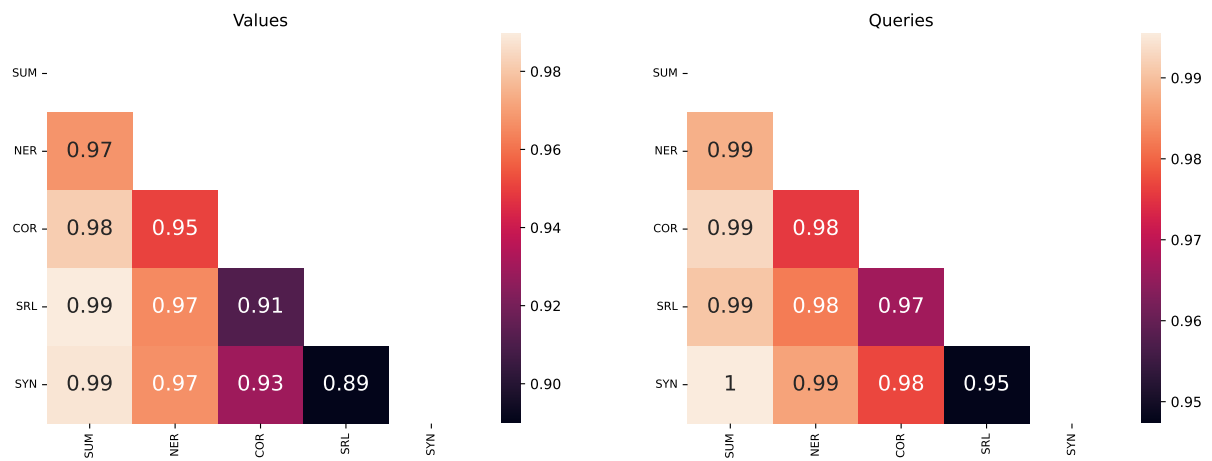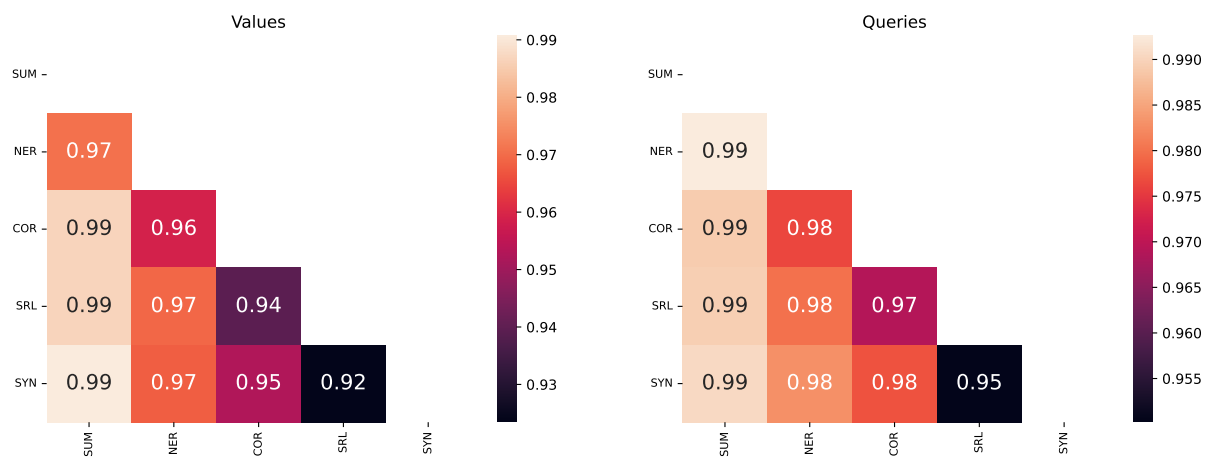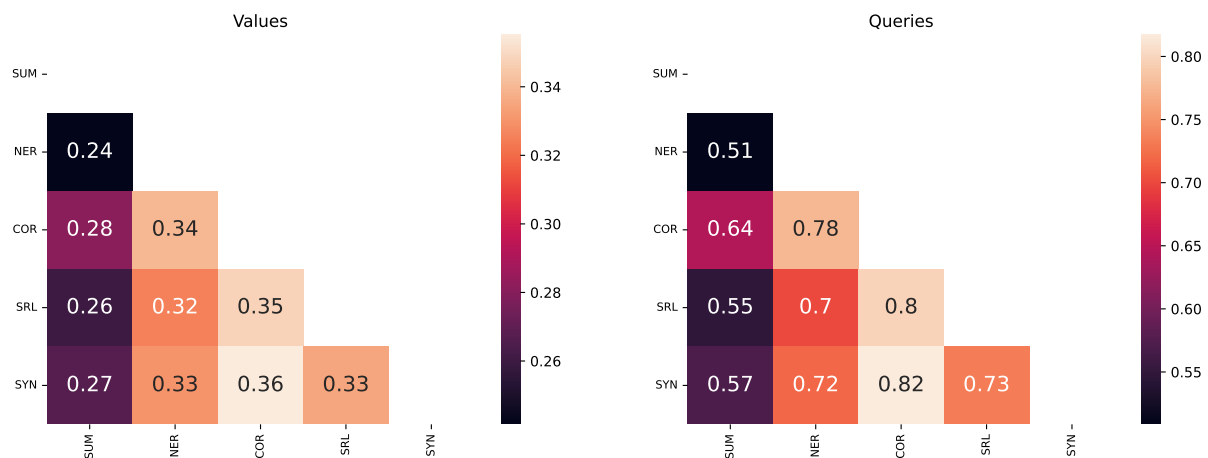

(b) Cosine Mistral Base

(c) L_2 Mistral Base

Figure 11: Mistral Base distance heat maps
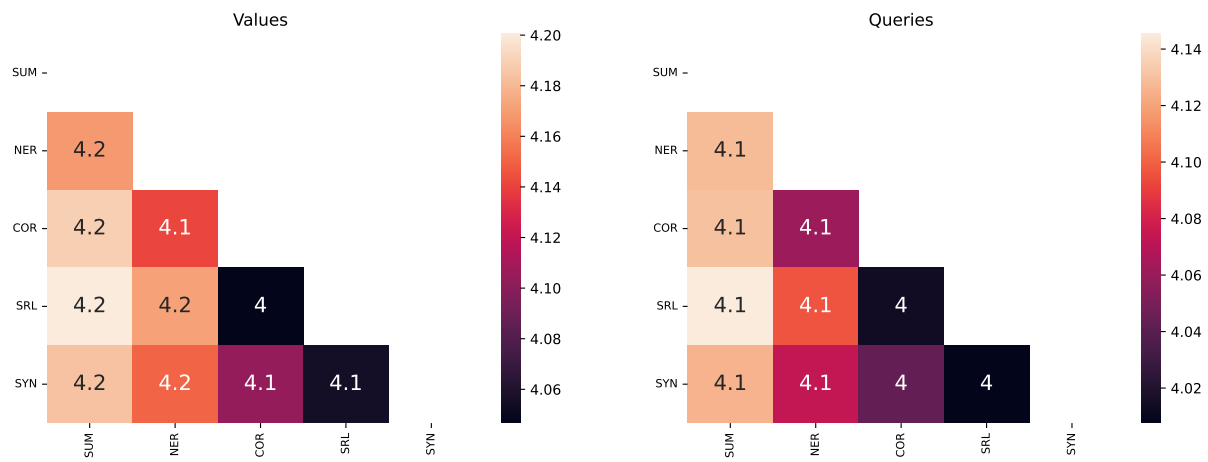
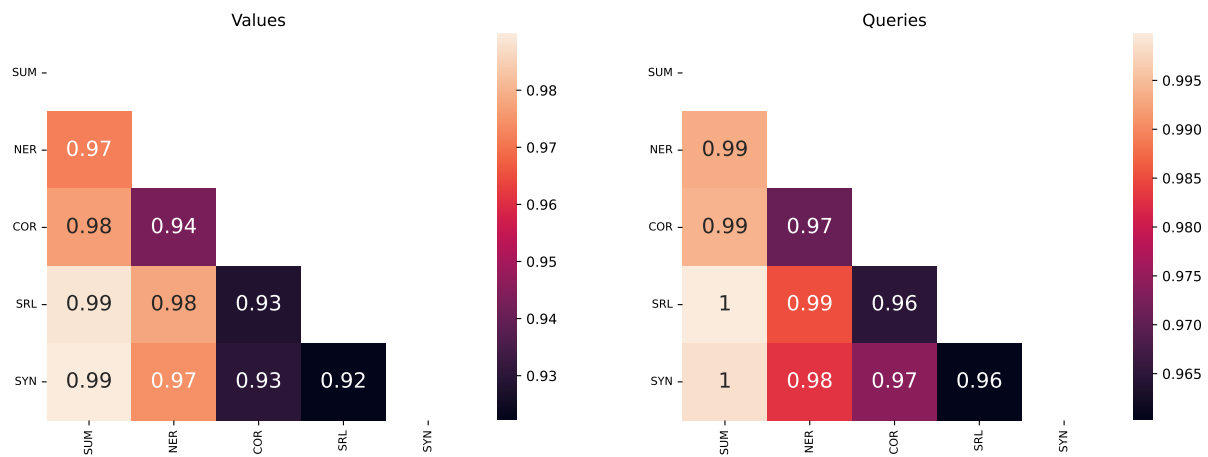(a) Grassmann Mistral Instruct

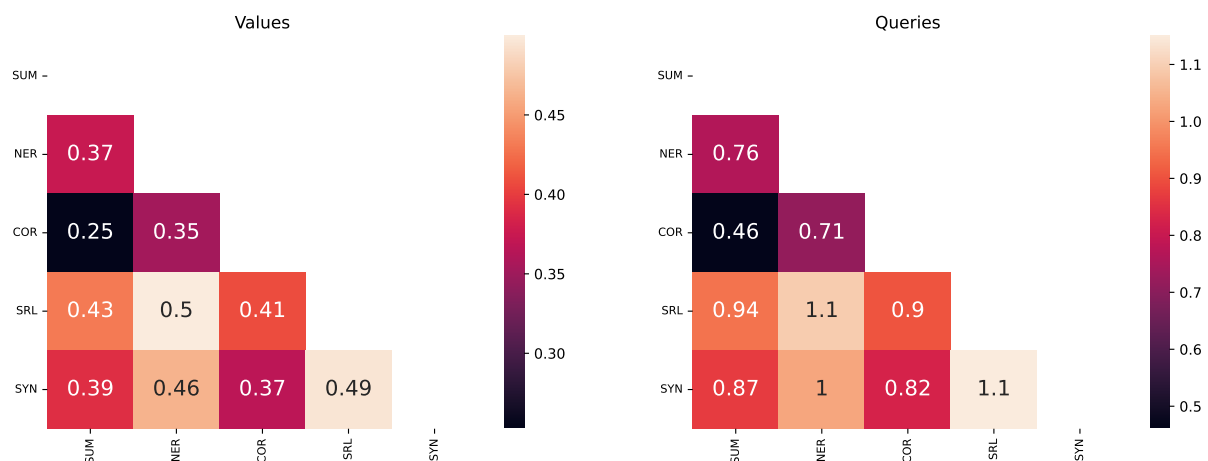(b) Cosine Mistral Instruct

(c) L_2 Mistral Instruct

Figure 12: Mistral Instruct distance heat maps
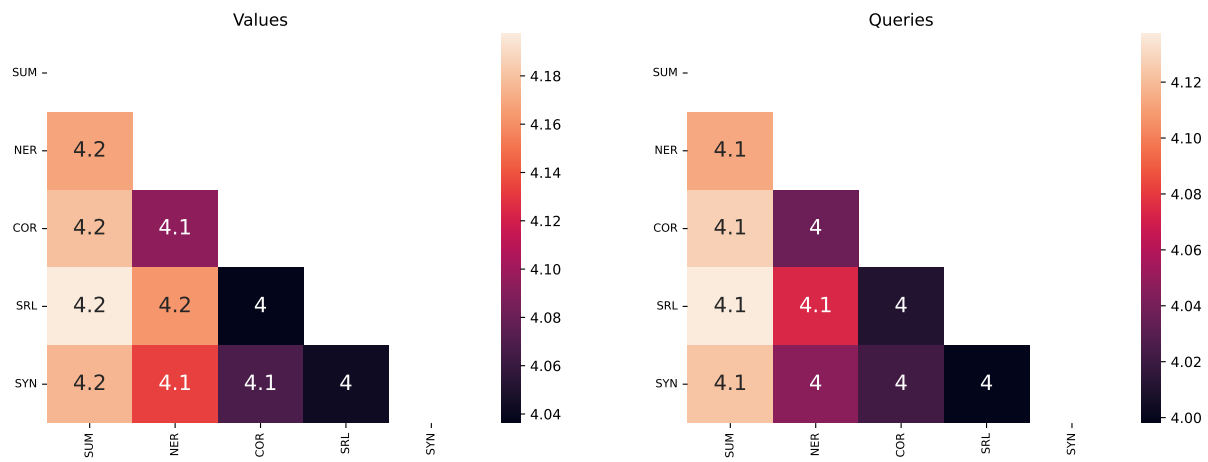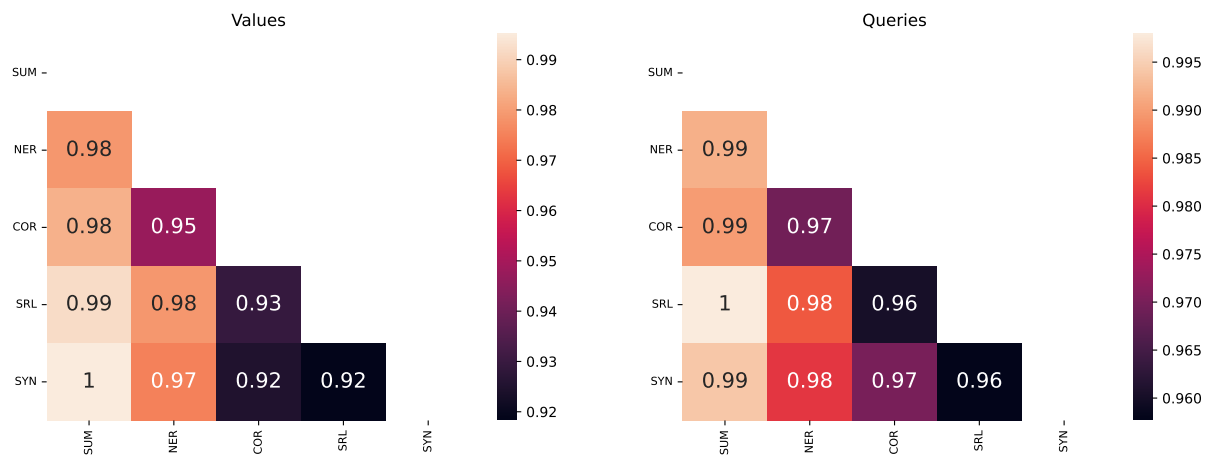
(a) Grassmann Llama 3 Base

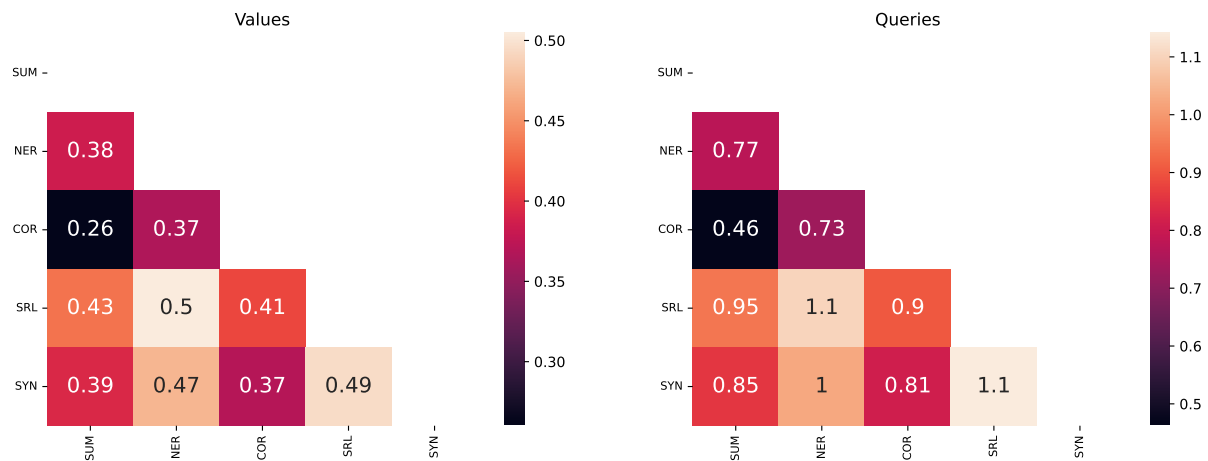(b) Cosine Llama 3 Base

(c) L_2 Llama 3 Base

Figure 13: Llama 3 Base distance heat maps

(a) Grassmann Llama 3 Instruct

(b) Cosine Llama 3 Instruct

(c) L_2 Llama 3 Instruct

Figure 14: Llama 3 Instruct distance heat maps