Detecting and Translating Language Ambiguity with Multilingual LLMs

Behrang Mehrparvar
University of Amsterdam
behrang.mehrparvar@student.uva.nl

Sandro Pezzelle
ILLC
University of Amsterdam
s.pezzelle@uva.nl

Abstract

Most languages could be ambiguous, which means the same conveyed text or speech, results in different actions by different readers or listeners. In this project, we propose a method to detect the ambiguity of a sentence using translation by multilingual LLMs. In particular, we hypothesize that a good machine translator should preserve the ambiguity of sentences in all target languages. Therefore, we investigate whether ambiguity is encoded in the hidden representation of a translation model or, instead, if only a single meaning is encoded. In our experiments, we have been able to predict the ambiguity of sentences with high accuracy using machine translation without direct use of semantics and only based on the reconstruction error of a function that maps the forward and backward translation hidden representations to each other. The potential applications of the proposed approach span i) detecting ambiguous sentences, ii) fine-tuning existing multilingual LLMs to preserve ambiguous information, and iii) developing AI systems that can generate ambiguity-free languages when needed.

1 Introduction

Language ambiguity is defined as the potential of different actions as a response to a single text by different people, based on their interpretations (Ceccato et al., 2004). This definition aligns with the semantic, syntactic, pragmatic tests along with identity tests defined in (Zwicky and Sadock, 1975) to identify ambiguous sentences.

Several research studies have been focusing on the ambiguity of language. For a comprehensive review on resolving ambiguities in NLP, refer to (Yadav et al., 2021). (Wang, 2011) have studied lexical and syntactic ambiguity in the Korean language. They proposed adding new words as a solution for lexical and syntactic ambiguities. (Ceccato et al., 2004) proposed a prototype for an ambiguity identification tool. They defined sentence ambiguity of a sentence, as a function of number of senses of each word in that sentence. Furthermore, Yadav et al. (2021) have proposed a comprehensive taxonomy of different types of language ambiguities.

Ambiguous	Disambiguation
“Give me the bat!” (Lexical)	“Give me the baton!”
“The professor said on Monday he would give an exam" (Syntactic)	“The professor said that on coming Monday he would give an exam"
“Jane saw the man with a telescope” (Semantic)	“Jane saw the man by using a telescope”
“I like you too!” (Pragmatic)	“I like you too like others do!”
“The prof said she would give us all A’s.” (Vagueness)	“The prof said the TA would give us all A’s.”
“Proposal” to “voorstel” and “aanzoek” (Translational)	“Research proposal”

Table 1: Various types of language ambiguity (Yadav et al., 2021) and their disambiguated versions.

In many languages including English, sentences do not always correspond to a unique set of possible behaviors and actions by different readers/listeners, which as we define, leads to language ambiguity. Table 1 lists different types of language ambiguities based on (Yadav et al., 2021), including examples and their disambiguated versions.

Language ambiguity brings up misunderstandings and conflicts in real-world interactions such as political, commercial, and cultural interactions (Bowe et al. (2014), Bachmann-Medick (1996)). This misunderstanding can lead to either wasting of huge amount of time in negotiation between the

Proceedings of the Fourth Workshop on Multilingual Representation Learning (MRL 2024), pages 310–323

parties for conflict resolution or even in the worst case results in conflicting actions (Kimmel (2006)). By using the powerful tools in NLU and NLP using language models, it could be possible to solve these issues.

The main research questions being investigated in this project are:

Question 1: *Do state-of-the-art Transformer-based MT models properly encode whether a sentence in the source language is (non-) ambiguous?*

Question 2: *Are both semantic validity and ambiguity preserved by the translation of these models, when the sentence is translated into a target language, and then translated back?*

Question 3: *Can we predict the ambiguity of a sentence by translating it into another language looking at the learned hidden representations?*

The main contribution of this work is proposing a solution that detects ambiguous sentences in different typos, without direct use of semantics. Furthermore, through our experiments, we conclude that ambiguity of the sentences are preserved in the hidden representation of the multilingual LLM translation model.

2 Related work

Before explaining the proposed approach, we review the related literature, consisting of ambiguity in NLP, ambiguity in machine translation, and an overview of multilingual LLMs.

2.1 Ambiguity in machine translation

Language ambiguity is a key aspect explored in machine translation (Baker et al. (1994), Jaspaert (1984)).

With the goal of disambiguation in translation, in Baker et al. (1994), the authors propose a source language analyzer component in their machine translation system that incorporates a controlled lexicon, a controlled grammar, and a semantic domain model.

One of the key points in dealing with ambiguity in translation is choosing the representation of the ambiguous sentence. The way we represent the sentence, directly influences the method we propose to detect ambiguity and/or disambiguate the sentence. Emele and Dorna (1998) suggest using a form of hierarchical recursive representation similar to a syntactic tree, to preserve the ambiguities between source and target language. In cases where the target language cannot preserve the ambiguity, the authors propose local disambiguation by asking the human user to specify the correct intention of the source sentence. In Boguslavsky et al. (2005), the authors propose a rule-based machine translation system that use a morphological structure and dependency tree structure to interactively disambiguate sentences.

Apart from syntactic structures, lexical representation of sentences is also crucial in disambiguation. In Sammer et al. (2006), the authors propose using human assistance in lexical ambiguity resolution in machine translation. They develop a system composed of a controlled language lexicon composed of words, word senses, their translations, and a short, intuitive gloss or set of clue words to help the user select the correct word sense during interaction with the machine translation system. Měchura (2022) investigates gender, number, and formality ambiguities in translation. In these cases, according to the paper, the machine translator either decided on a random or statistically biased translation which requires to ask the human the right questions to disambiguate the text manually.

Unlike Baker et al. (1994), our method is not rule-based and hard-coded which results in a more flexible ambiguity detection method. Also, contrary to Sammer et al. (2006), we do not require a predefined lexicon for detecting ambiguous words. Unlike Emele and Dorna (1998) and Boguslavsky et al. (2005),our approach however represents the sentences in forms of vector representations in the LLM but still do not directly rely on these representations in detecting ambiguity.

In this project, we do not provide direct solutions for disambiguation. As of future work, similar to Měchura (2022), our method can be considered as a human-assisted machine translation (HAMT) solution defined in Alzeebaree (2020) which the user is asked to disambiguate detected ambiguous sentences in the input text. Also, the machine translation model we use is trained based on the interlingua approach.

2.2 Ambiguity in the Era of LLMs

Language ambiguity, as a subset of semantic underspecification (Egg, 2010) which is introduced as the possibility for a linguistic signal to convey only part of the information needed for communication to succeed ((Hada et al., 2023)).

Liu et al. (2023) have proposed a benchmark for evaluating pre-trained language models to recognize ambiguity and disentangle possible meanings. They capture the ambiguity of the sentences through their entailment relations with other sentences. They have covered different ambiguity types including pragmatic, lexical, syntactic, scopal, coreference, figurative, and other ambiguities. Based on their benchmark, they realized that disambiguation of sentences using state-of-the-art LLMs is still very challenging.

More recently, in Wildenburg et al. (2024), the authors use perplexity measures to identify underspecified sentences from the pairs in their proposed DUST dataset. Based on Egg (2010), they define four types of underspecified sentences.

In (Pezzelle, 2023) the author has investigated how multi-modal models deal with semantic underspecification and how communicative approaches would provide solutions to this type of task. In Hutchinson et al. (2022), the authors also investigated semantic underspecification in text used to generate images. They studied a taxonomy of the family of multi-modal tasks and provided a list of risks and concerns regarding ambiguity in multimodal text and image tasks.

Our work builds on this previous research investigating how LLMs deal with ambiguity. However, we make a step further, and consider how ambiguity is represented by current models across various languages. To the best of our knowledge, ours is the first work studying ambiguity in multilingual LLMs.

2.3 Multilingual Large Language Models

With the advent of Transformer-based language models, multilingual models have been proposed. These models are trained with data from many languages and can perform machine translation among many other NLP tasks with higher performance, compared to traditional approaches (Liu et al. (2024), Liao et al. (2024)).

As multilingual LLMs are trained on data from multiple languages, the mechanism of how these models perform certain tasks has been recently studied. Knowing the internal mechanism could provide us insight into the ambiguity encoded in the representation of the hidden layers of the LLM.

Choenni et al. (2023) have studied how individual languages in multilingual LLMs benefit from each other as in cross-lingual sharing at the data level. They found that multilingual LLMs rely on data from multiple languages during fine-tuning which can be useful in real-world translation models. Furthermore, in Zhang et al. (2023), the authors studied how knowledge transfer happens in multilingual LLMs during translation while limited multilingual training data leads to advanced multilingual capabilities. According to their finding, LLMs struggle to provide accurate results in translation-variant tasks. Liu et al. (2024) have studied the connections of multilingual activation patterns in LLMs at the level of language families. Similar to Tang et al. (2024), they have discovered (non-)language-specific neurons in the LLMs which capture meanings, regardless of specific target language.

Finally, Zhao et al. (2024) have studied the representation of multilingual LLMs across the layers of the model and realized that the first layers understand the questions by converting the multilingual input to English, the intermediate layers perform problem-solving, mainly in English, and in the last layers, the models generate the response according to the original language. Knowing the outcome of their results in finding the responsibility of different layers of multilingual LLMs could help us choose the representation of the right layer for our experiments.

In Qi et al. (2023), the authors study the cross-lingual consistency of factual knowledge and propose a metric to evaluate knowledge consistency across languages independently from accuracy. Tanwar et al. (2023) study cross-lingual in-context learning.

Finally, Zhu et al. (2023), (Zhu et al., 2024) and Gao et al. (2024) have studied multilingual machine translation in LLMs. Through their approach, they where able to improve zero-shot translation performance by learning language-agnostic representations in the multilingual LLMs.

3 Proposed method

In this project, we aim at testing how language ambiguity is represented in multilingual LLMs. We propose language translation as an action performed by LLM agents. Accordingly, we propose a four-step approach in detecting language ambiguity, as illustrated in figure 1:

1. **Translation**: Translate the input text from the source language into the target languages using a multilingual LLM. Then extract the hidden representation from the LLM.

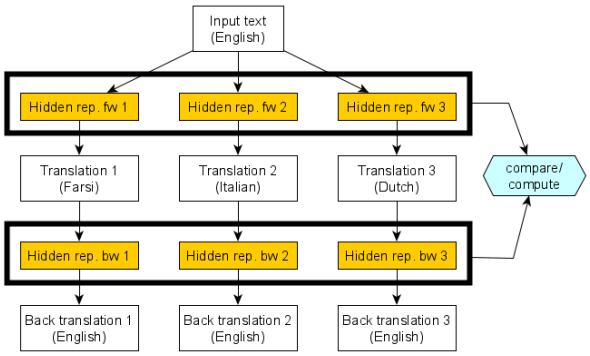


Figure 1: Proposed approach in language ambiguity detection using LLM translation consisting of four steps: 1) translating the text into the target languages, 2) translating back the new texts into the source language, 3) comparing the pairwise representations, 4) computing the overall measure of ambiguity.

2. **Back-translation**: Translate back the output texts of the first step from the target language into the source language using the same LLM. Then extract the hidden representation of the state of the LLM as a vector.

3. **Mapping function**: Compute a function that maps the two representations above. Note that due to the both complexity of the LLM and also various types of information stored in the representations such as semantics, syntax, language information, etc., we do not expect an identity function to be able to map the representations, even in case of unambiguous sentences.

4. **Ambiguity evaluation**: Compute an overall measure of ambiguity based on the properties of the mapping function. We hypothesize that the mapping function learns high-level feature encoding how ambiguous a sentence is, independently of its meaning. Therefore, we can use features of this mapping function to quantify how much ambiguity was preserved in the translation and back-translation.

Considering n different meanings for input text t_A and m different interpretations of the output text t_B, in the worst case we would have $n \times m$ different translation meaning pairs, which complicates the problem of ambiguity in translation. As it has been noted in section 4.3, the translation process by itself can be a source of ambiguity.

The LLM works as a function $f(.)$ defined in equation (1):

$$r \mapsto f(t, l_s, l_t) \tag{1}$$

where t is the input text, l_s is the source language, l_t is the target language, and r is the vector representation of the hidden state of the LLM. By applying the translation function $f(.)$ in steps 1 and 2 listed above, the representation vectors can be found as in equation (2):

$$\begin{aligned} r_A &= f(t_A, l_1, l_2) \\ r_B &= f(t_B, l_2, l_1) \end{aligned} \tag{2}$$

where t_A is the input text and t_B is the generated output text from the translation using the LLM in step 1.

The hidden representation r consists of a distributed representation of multiple factors, not only including the semantics (Bau (2022), Zhang (2024)) and it is not easy to simply disentangle these factors and manually extract the representation of the input text t from the representation r. Also as the representation r contains factors such as the information about the source and target language, the translation task, etc., we can not directly compare the two representations r_A and r_B to detect ambiguity in the text. Therefore we propose a different approach in detecting ambiguity.

In the first step, we define a function $g(.)$ that maps the two representations to each other as illustrated in equation (3):

$$r_B = g(r_A) \tag{3}$$

where r_A and r_B are the representations found from equation (2) and $g(.)$ is the mapping function.

To find the function $g(.)$, we learn a simple auto-encoder with a single hidden layer of size s_H, input size of s_A and output size of s_B. Note that as the translation in steps 1 and 2 are both performed using the same LLM, we have $s_A = s_B$.

The auto-encoder maps the input translation representation r_A to the output translation representation r_B. The error of the network implementing $g(.)$ is defined as the normalized mean squared error (NMSE) of the elements of the two representations r_A and r_B (the actual equations can be found in the Appendix A).

We define the function $c(.)$ as complexity of the function $g(.)$ as follows:

$$c(g) = s_H / s_A \quad (4)$$

where s_H and s_A are the sizes (number of neurons) in the hidden layer H and input r_A of the neural network implementing the $g(.)$ function.

By learning function $g(.)$, for each text t_A in the input dataset, we can evaluate the translation error $e(.)$ for each setting of the network complexity $c(.)$ with different hidden layer sizes. Figure 2 reports the error of the function against its complexity.

The main idea for using an auto-encoder is based on the assumption that: (1) We expect the auto-encoder will behave differently for ambiguous vs unambiguous sentences; (2) in particular, we conjecture that model size and the target language will affect differently the model when dealing with ambiguous vs unambiguous sentences.

We propose using a simple neural network model to predict ambiguity using the data points in the elbow chart in figure 2 as input in a supervised manner.

3.1 Experiments

The Dataset of semantically Underspecified Sentences by Type (DUST)[1] contains a balanced number of ambiguous and unambiguous English sentences. We use a multi-language translation model such as Facebook M2M100[2] (Fan et al., 2020) to translate each sentence from English to other possible languages and translate them back to English. The model is trained on any pairs of 100 languages in a supervised manner with 15.4B parameters has resulted a high performance compared to English-Centric approaches. The pairs of sentences are selected from different sources mentioned in (Fan et al., 2020). The scope of the paper is to study ambiguity detection in LLM translation for the first time, therefore we chose one model not necessarily the state-of-the-art. Therefore, future work should indeed compare various models. We consider German, Greek, Persian, Spanish, French, Hindi, Italian, Korean, Dutch, Russian, Turkish, Croatian, Romanian and Chinese as our target languages. After translation, we extract the hidden states of the LLM for the two translation steps as defined in equation (5):

$$T_A = \{t_A^j\}, R_A = \{r_A^j\}, R_B = \{r_B^j\} \quad (5)$$

[1]https://github.com/frank-wildenburg/DUST

[2]https://huggingface.co/facebook/m2m100_418M

After learning the network for the function $g(.)$, we feed all the r_A's to the network and capture the outputs r'_B's. Using equations (4) and (6), we find the complexity and error for each sample and each network size. Figure 2 shows the elbow for the mapping functions of an ambiguous sentence and its unambiguous version.

For classification, we used either a neural network or a logistic regression model. Further details about the classification experiments are explained in section 4.2.

3.1.1 Qualitative Analysis

As an analysis of the experiment before, for the misclassified samples, the two authors of the paper, who are proficient in two languages (Farsi and Italian) out of the set reported above, verified if the corresponding sentence in the target language is (A) semantically valid and (B) (non-)ambiguous. Semantic validity is verified by asking the human user whether the sentence is correctly translated, and ambiguity is verified by asking whether the translated sentence is (still) ambiguous or not.

3.2 Evaluation

We translate ambiguous and unambiguous English sentences to the languages listed above and investigate whether the meaning has changed through analysis of the hidden states of the multilingual LLMs.

Based on our evaluation protocol, if we obtain high accuracy in predicting the ambiguity of ambiguous sentences, we can conclude that the model is able to properly encode ambiguity in its hidden representations (research question 1). Furthermore, the high accuracy shows that predicting ambiguity using multilingual LLM translation models is possible (research question 3).

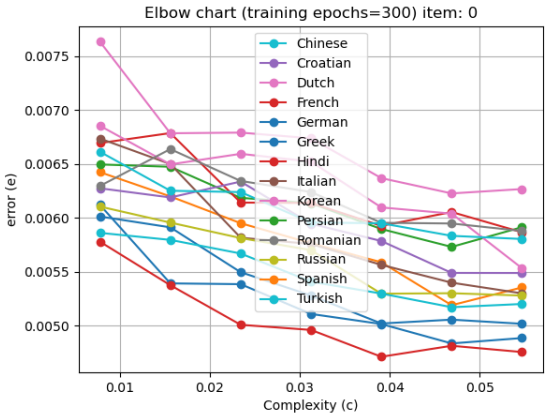
Human error analysis will help us shed light on the research question 2.

4 Results

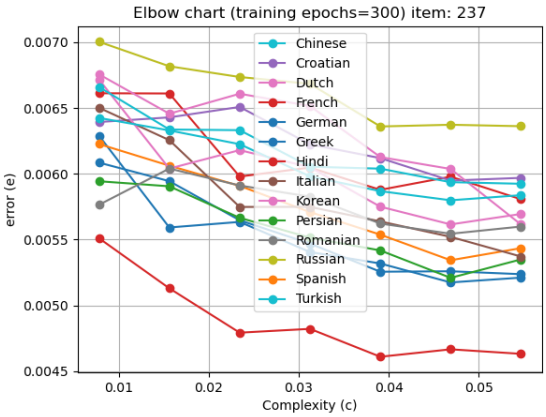
In this section, we provide the results of our experiments.

4.1 Discriminability

In the first step of our analysis, we examined the discriminability of reconstruction error of the best auto-encoder per each language in predicting ambiguity of the sentences. Figure 5 illustrates the distribution of reconstruction error along languages for



(a) "Andrei picked up the chair or the bag and the telescope" (ambiguous)



(b) "Andrei picked up the chair, or both the bag and the telescope" (unambiguous)

Figure 2: Illustration of the mapping function for an ambiguous sentence and its unambiguous version.

Language	t-test	p-value
German	-0.341	0.33
Greek	0.510	0.610
Persian	-1.95	0.051
Spanish	-0.087	0.931
French	-1.072	0.285
Hindi	1.828	0.069
Italian	-0.821	0.413
Korean	1.864	0.063
Dutch	-2.253	**0.025**
Russian	-0.905	0.366
Turkish	-1.557	0.121
Croatian	-1.034	0.452
Romanian	-1.594	0.112
Chinese	-3.307	**0.001**

Table 2: T-test statistics indicating discriminability of reconstruction error of best auto-encoder for ambiguity. We test significance at $pvalue < 0.05$.

each class. To evaluate the discriminability, we performed t-test statistics by verifying pseudo-normal distribution of data. The detailed results are listed in table 2.

Based on the t-test results, we can conclude that mean reconstruction errors for separate target languages are not informative enough to discriminate ambiguous and unambiguous sentences, except for a limited number of languages.

4.2 Classification

To determine the most informative variables for classification, we performed several experiments, each including a different setting composed of the options listed in Appendix B.

Table 3 shows the results of classification in all experiment settings. The detailed analysis of the findings for these experiments is provided in section 5.

4.3 Source of Ambiguity

After classifying the data, we investigated the source of misclassification using annotation for the Italian and Persian languages. Accordingly, we found both machine translation and also the incapability of the target language itself in preserving the ambiguity, as the sources of misclassification. We only performed a preliminary and arguably limited annotation, but in future work we chould recruit many more participants and conduct a much larger-scale human analysis. Figure 3 illustrates these results.

From the misclassified sentences (examples shown in table 6), considering two target languages (Italian and Persian) we found the following outcomes:

- Ambiguity was lost in 44.68% of the Italian and 51.02% of the Persian target sentences (out of misclassified ambiguous sentences).
- From the misclassified sentences that the ambiguity was lost, in the Italian target language, 85.71% of the loss was because of the translation model and the sentence could be written in an ambiguous sense by a native human. However, none of the loss of ambiguity was because of the translation in the Persian target language and the native Persian human was

Input	Input variable	Output	Model	Accuracy	F-Measure
Persian	Differences	Amb. Vs unamb.	LR	57.81%	0.578
Best AE	Values	Amb. Vs unamb.	LR	66.67%	0.667
Along languages	Differences	Amb. Vs unamb.	LR	85.87%	0.859
Whole	Differences	Amb. Type	LR	92.83%	0.928
Whole	Differences	Amb. Vs unamb.	LR	88.19%	0.882
Best AE	Values	Amb. Vs unamb.	NN	73.21%	0.732
Whole	Values	Amb. Vs unamb.	NN	81.99%	0.820
Whole	Values	Amb. Type	NN	78.26%	-
Whole	Differences	Amb. Type	NN	93.04%	0.925
Whole	Differences	Amb. Vs unamb.	NN	94.94%	0.949

Table 3: Classification results for different settings. For classifying ambiguous vs unambiguous sentences the chance level accuracy is 50.0% and for ambiguity type it is 36.58%

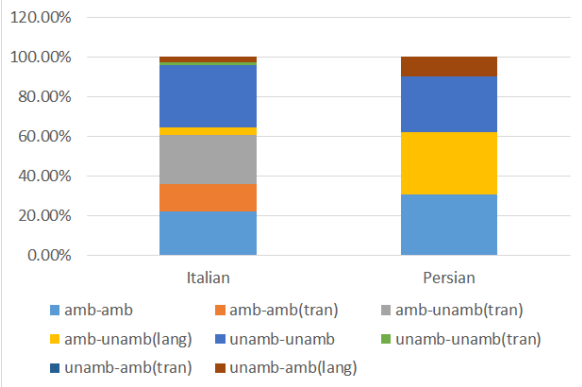


Figure 3: Misclassified samples distribution - format: source-target(problem): *amb*: ambiguous, *unamb*: unambiguous, *tran*: source of misclassification is wrong machine translation, *lang*: source of misclassification is target language incapability in transferring ambiguity.

also unable to translate the ambiguity into the target language due to the innate difference between English and Persian languages.

- From the unambiguous misclassified sentences, in 7.69% of the cases, ambiguity was introduced in Italian translation, none of which was because of wrong translation by the machine, but because of the innate difference between the target language and English. This percentage increases in Persian to 26.67% of the unambiguous misclassified sentences which was similarly due to the innate difference in languages and not because of machine translation.

- We can conclude that 68.49% of the misclassified sentences in total were correctly translated in terms of ambiguity in Italian while 58.23% in Persian, from which 78.26% (for Italian) and 0.0% (for Persian) was because of a machine translation problem.

5 Discussion

Based on the results of our classification experiments shown in table 3, we achieved the following findings:

1. Single language translation is not informative enough in predicting ambiguity. By moving from one language (Persian) to all languages, we achieved 85.87% accuracy (from 57.81%). This could be due to the effect of adding more informative input features (information about other language translations) to the classification algorithm.

2. Single best auto-encoder is not informative enough in predicting ambiguity. The accuracy has changed from 66.67% to 88.19% by introducing more auto-encoder models even with lower complexities. Adding more features about the gradual change over the complexity of the auto-encoder model could explain this phenomenon.

3. Adding reconstruction error differences between languages improves accuracy. By adding this information we achieved 88.19% accuracy compared to 85.87%. Accordingly, adding more features about the properties of the mapping function mesh improved the accuracy.

4. Reconstruction error differences is more informative than their values. These phenomena can be observed from the results by improving from 81.99% to 94.94% accuracy. We

can conclude that the shape of the mapping function is informative not the position of it. However, we would expect that a nonlinear complex classifier would also be able to pick this feature.

5. A simple linear model can perform relatively close to a complex neural network model. The accuracy of the complex model was 94.94% compared to 88.19% for the linear model. Learning more complex and nonlinear features actually helped the classification.

6. Predicting more detailed classes improves the accuracy in linear models. For the linear model, the accuracy have changed from 88.19% (F-measure 0.820) to 92.83% (F-measure 0.928) by changing to multi-class classification. It can be explained by classifying more detailed regions in the misclassified regions. For more details on the distribution of the classes along the main two principle components, refer to figure 6. For the neural network however, the classification result decreased from 94.94% to 93.04% by moving to multi-class classification. Compared to the increase of accuracy in the linear model, we can explain that the neural networks have been already able to learn the nonlinear boundaries in the input space and already got a high accuracy in two-class classification.

Moving back to our initial research questions, based on the results in table 3, we can claim that it is possible to predict sentence ambiguity using machine translation. However, we can not claim that the semantic validity and ambiguity is preserved by translation for all target languages and it highly depends on the language. Finally, we conclude that the ambiguity of the sentence is actually encoded in the hidden representation of the LLMs, as the ambiguity is predictable from these representations.

The main contribution of the project is predicting ambiguity of the sentences, without direct use of semantics. As explained in section 3 this feature is achieved by classifying the ambiguity based on the shape of the mapping function. As a consequence, the algorithm does not require extensive training data to cover the whole semantic. Furthermore, the approach is potentially much more generalizable to unseen sentences with unseen semantics. Also, the model would be robust to changes to the input distribution as it is independent of the semantics.

6 Future work

One future direction method is to investigate in more details the source of misclassification for all fourteen target languages other than Italian and Persian. Other than that, detecting the source of ambiguity in sentences in terms of words could be an interesting direction. Furthermore, extending the method to different source languages other than English could also be considered as future work.

One of the potential applications of an ambiguity detection method could be in automatic translation of critical documents e.g. legal, political, commercial, where the user is asked to clarify the ambiguity of the source language manually, to prevent misunderstanding and potential conflicts.

Fine-tuning existing multilingual large language models to preserve ambiguity in sentences could be another potential application of the proposed method.

Finally, the trained classifier model can potentially be used as a partial loss function for designing and optimizing ambiguity-free AI-generated human languages investigated at Synaptosearch[3]. In order to do so, for each input sentence generated by the AI, the ambiguity is measured using the model and the gradient with respect to the input is calculated and used to optimize the loss function term related to ambiguity.

Ambiguity can be considered of a strength of the language in cases such as providing efficient means of communication or when it is used as amphibology in literature. However, in critical political, commercial and cultural cases and social media, unintended ambiguity results in misunderstandings and conflicts. The outcome of the misunderstanding could lead to spending a lot of time in negotiation to elaborate the meaning, or in worse case conflicting actions.

One major organization that can benefit from the proposed research is the United Nations (UN) where different countries with different languages interact with each other. Considering automatic translation in such organizations where a speech/text is translated into many languages, detecting and informing the potential ambiguities to both the speaker/writer and the listener/reader, would prevent potential misunderstandings, tedious negotiations, and conflicting actions between the nations and parties in the long term (Bowe et al. (2014), Kimmel (2006)).

[3]https://synaptosearch.com/

References

Yaseen Alzeebaree. 2020. Lexical and Structural Ambiguity in Machine Translation: An Analytical Study. *Eastern Journal of Languages, Linguistics and Literatures*, 1(1).

Doris Bachmann-Medick. 1996. Cultural misunderstanding in translation: Multicultural coexistence and multicultural conceptions of world literature. *Erfurt Electronic Studies in English*, 7(1996):1–15.

Kathryn Baker, Alexander Franz, Pamela Jordan, Teruko Mitamura, and Eric Nyberg. 1994. Coping with ambiguity in a large-scale machine translation system. In *COLING 1994 Volume 1: The 15th International Conference on Computational Linguistics*.

Anthony Bau. 2022. *Interactions Between Syntax and Semantics in Language Models*. Ph.D. thesis, Massachusetts Institute of Technology.

Igor M Boguslavsky, Leonid L Iomdin, Alexander V Lazursky, Leonid G Mityushin, Victor G Sizov, Leonid G Kreydlin, and Alexander S Berdichevsky. 2005. Interactive resolution of intrinsic and translational ambiguity in a machine translation system. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 388–399. Springer.

Heather Bowe, Kylie Martin, and Howard Manns. 2014. *Communication across cultures: Mutual understanding in a global world*. Cambridge University Press.

Mariano Ceccato, Nadzeya Kiyavitskaya, Nicola Zeni, Luisa Mich, and Daniel M Berry. 2004. Ambiguity identification and measurement in natural language texts. Publisher: University of Trento.

Rochelle Choenni, Dan Garrette, and Ekaterina Shutova. 2023. How do languages influence each other? Studying cross-lingual data sharing during LLM fine-tuning. *arXiv preprint arXiv:2305.13286*.

Markus Egg. 2010. Semantic underspecification. *Language and Linguistics Compass*, 4(3):166–181. Publisher: Wiley Online Library.

Martin C Emele and Michael Dorna. 1998. Ambiguity preserving machine translation using packed representations. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 365–371.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Edouard Grave, Michael Auli, and Armand Joulin. 2020. Beyond english-centric multilingual machine translation. *Preprint*, arXiv:2010.11125.

Pengzhi Gao, Zhongjun He, Hua Wu, and Haifeng Wang. 2024. Towards boosting many-to-many multilingual machine translation with large language models. *arXiv preprint arXiv:2401.05861*.

Rishav Hada, Varun Gumma, Adrian de Wynter, Harshita Diddee, Mohamed Ahmed, Monojit Choudhury, Kalika Bali, and Sunayana Sitaram. 2023. Are large language model-based evaluators the solution to scaling up multilingual evaluation? *arXiv preprint arXiv:2309.07462*.

Ben Hutchinson, Jason Baldridge, and Vinodkumar Prabhakaran. 2022. Underspecification in scene description-to-depiction tasks. *arXiv preprint arXiv:2210.05815*.

Lieven Jaspaert. 1984. About the treatment of ambiguity in machine translation. *ITL-International Journal of Applied Linguistics*, 64(1):1–21.

Paul R Kimmel. 2006. Culture and conflict. *The handbook of conflict resolution: Theory and practice*, pages 625–648.

Baohao Liao, Christian Herold, Shahram Khadivi, and Christof Monz. 2024. Ikun for wmt24 general mt task: Llms are here for multilingual machine translation. *arXiv preprint arXiv:2408.11512*.

Alisa Liu, Zhaofeng Wu, Julian Michael, Alane Suhr, Peter West, Alexander Koller, Swabha Swayamdipta, Noah A Smith, and Yejin Choi. 2023. We're afraid language models aren't modeling ambiguity. *arXiv preprint arXiv:2304.14399*.

Weize Liu, Yinlong Xu, Hongxia Xu, Jintai Chen, Xuming Hu, and Jian Wu. 2024. Unraveling Babel: Exploring Multilingual Activation Patterns within Large Language Models. *arXiv preprint arXiv:2402.16367*.

Michal Měchura. 2022. A taxonomy of bias-causing ambiguities in machine translation. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 168–173.

Sandro Pezzelle. 2023. Dealing with semantic underspecification in multimodal NLP. *arXiv preprint arXiv:2306.05240*.

Jirui Qi, Raquel Fernández, and Arianna Bisazza. 2023. Cross-Lingual Consistency of Factual Knowledge in Multilingual Language Models. *arXiv preprint arXiv:2310.10378*.

Marcus Sammer, Kobi Reiter, Stephen Soderland, Katrin Kirchhoff, and Oren Etzioni. 2006. Ambiguity reduction for machine translation: Human-computer collaboration. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 193–202.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. Language-Specific Neurons: The Key to Multilingual Capabilities in Large Language Models. *arXiv preprint arXiv:2402.16438*.

Eshaan Tanwar, Manish Borthakur, Subhabrata Dutta, and Tanmoy Chakraborty. 2023. Multilingual llms are better cross-lingual in-context learners with alignment. *arXiv preprint arXiv:2305.05940*.

William Shi-Yuan Wang. 2011. Ambiguity in language. *Korea Journal of Chinese Language and Literature*, 1:3–20.

Frank Wildenburg, Michael Hanna, and Sandro Pezzelle. 2024. Do Pre-Trained Language Models Detect and Understand Semantic Underspecification? Ask the DUST! *arXiv preprint arXiv:2402.12486*.

Apurwa Yadav, Aarshil Patel, and Manan Shah. 2021. A comprehensive review on resolving ambiguities in natural language processing. *AI Open*, 2:85–92. Publisher: Elsevier.

Xiang Zhang, Senyu Li, Bradley Hauer, Ning Shi, and Grzegorz Kondrak. 2023. Don't trust ChatGPT when your question is not in English: A study of multilingual abilities and types of LLMs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7915–7927.

Xulang Zhang. 2024. Disentangling syntactics, semantics, and pragmatics in natural language processing.

Yiran Zhao, Wenxuan Zhang, Guizhen Chen, Kenji Kawaguchi, and Lidong Bing. 2024. How do Large Language Models Handle Multilingualism? *arXiv preprint arXiv:2402.18815*.

Wenhao Zhu, Shujian Huang, Fei Yuan, Shuaijie She, Jiajun Chen, and Alexandra Birch. 2024. Question Translation Training for Better Multilingual Reasoning. *arXiv preprint arXiv:2401.07817*.

Wenhao Zhu, Hongyi Liu, Qingxiu Dong, Jingjing Xu, Lingpeng Kong, Jiajun Chen, Lei Li, and Shujian Huang. 2023. Multilingual machine translation with large language models: Empirical results and analysis. *arXiv preprint arXiv:2304.04675*.

Arnold M Zwicky and Jerrold M Sadock. 1975. Ambiguity tests and how to fail them. In *Syntax and Semantics volume 4*, pages 1–36. Brill.

A Normalized mean squared error

The network error is computed using Normalized mean squared error defined in equation (6):

$$
\begin{aligned}
e(r_A, r_B) &= \frac{1}{s_A} \sum_{i=0}^{s_A} \frac{(r_A^i - r_B^i)^2}{\overline{r_A}\,\overline{r_B}} \\
\overline{r_A} &= \frac{1}{s_A} \sum_{i=0}^{s_A} r_A^i \qquad (6) \\
\overline{r_B} &= \frac{1}{s_B} \sum_{i=0}^{s_B} r_B^i
\end{aligned}
$$

where r_X^i is the i'th element of representation r_X and s_X is the size (number of neurons) of r_X.

B Experiment settings

The experiment settings consisted of several options defined in table 4.

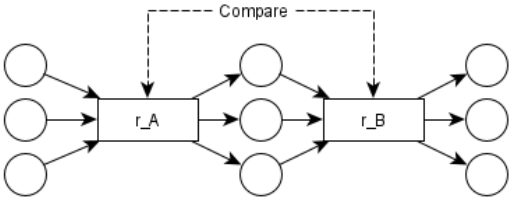
Setting	Options
Input type	- Single language across all auto-encoder models - All languages only for the best auto-encoder - Only relations across languages - Whole mapping functions
Input variable	- Reconstruction error - Reconstruction error difference
Output	- Ambiguous vs Unambiguous - Ambiguity type
Model	- Logistic regression - Neural network
Cross-validation	- 10-fold

Table 4: Experiment settings for ambiguity classification

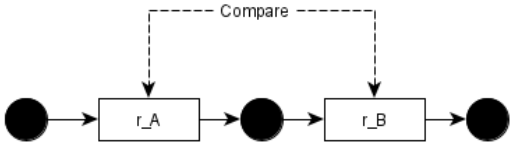
C Additional figures

Considering several possibilities of translating (un)ambiguous sentences, we summarize 6 states that can be found in table 5 and figure 4.

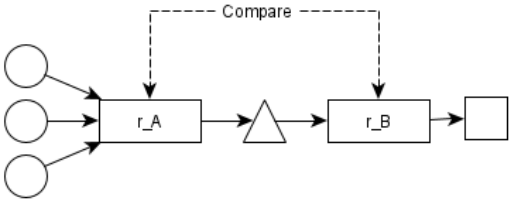
According to figure 4, for unambiguous sentences, state $sU0$ is desirable and for ambiguous source sentences, for all target languages, either of the states $sA0$ or $sA2$ is desirable. In other words, if a sentence is ambiguous, it should be either ambiguous in all target languages, or none of them.



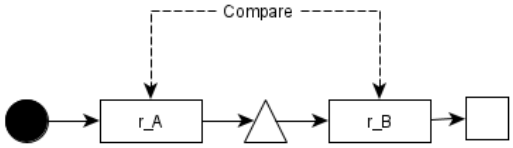
(a) State sA0



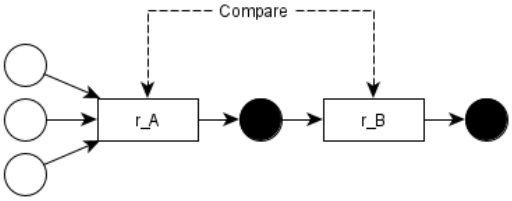
(b) State sU0



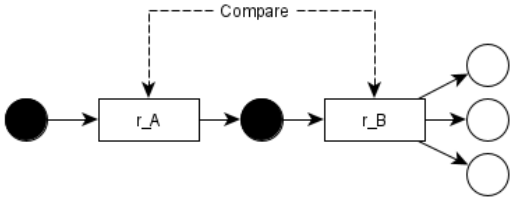
(c) State sA1



(d) State sU1



(e) State sA2



(f) State sU2

Figure 4: Possible states of the 2-step translation approach proposed in the project. White circles indicate certain meanings associated to an ambiguous sentence. Black circles indicate a biased meaning from possible meanings of an ambiguous sentence. Rectangles indicate the internal hidden states of a translation step. Triangles and squares indicate incorrect translations. For detailed description about the possible states refer to table 5.

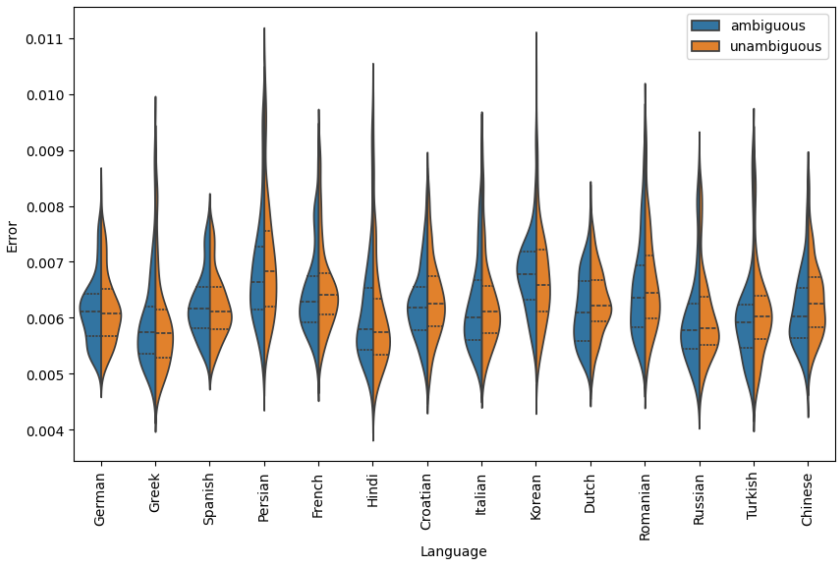


Figure 5: Discriminability of reconstruction error along language for the best auto-encoder. Languages other than Dutch and Chinese are not significantly separable according to the p-value in table 2.

Table 5: Possible states of the 2-step translation approach proposed in the project.

Tag	Source	Target	Case study	Hyp. Score	Notes
sA0	Ambiguous	Ambiguous	26%	0	Perfect hypothetical translation and rich target language. But score doesn't detect ambiguity.
sA1	Ambiguous	Incorrect	38%	1	Incorrect translation in step 1.
sA2	Ambiguous	Unambiguous	36%	?	If the score is 1, then we could conclude that ambiguity is encoded in the representation and the representation is not biased towards certain meanings. Also reaching this state might be because of the unambiguity in the target language by itself.
sU0	Unambiguous	Unambiguous	30%	0	
sU1	Unambiguous	Incorrect	70%	1	Only one sentence in case study; not reliable statistically.
sU2	Unambiguous	Ambiguous	0%	1	Very rare, but possible.

Input Text	Input ambiguity	Target language	Back translation	Back-translation ambiguity	Error state
Andrei and Danny moved the yellow bag and chair	Amb.	Persian	Andrew and Danny transferred the yellow bag and the chair.	Unamb.	sA2
Andrei and Danny held the green chair and bag	Amb.	Italian	Andrei and Danny have the green chair and the bag.	Unamb.	sA2
Andrei looked at Danny moving a yellow bag	Amb.	Persian	Andrew looked at Danny that the yellow bag was rolling around.	Wrong	sA1
Andrei held the bag, and either the telescope or the chair	Unamb.	Persian	Andrei kept the bag, or a telescope or a chair.	Wrong	sU1
Andrei picked up the chair, or both the bag and the telescope	Unamb.	Italian	Andrei took the chair, either the bag or the telescope.	Wrong	sU1
Danny moved the telescope that was on the bag	Unamb.	Persian	He moved the telescope on the bag.	Amb.	sU2
Danny left the chair while holding a green bag	Unamb.	Italian	Danny left the chair holding a green bag	Amb.	sU2

Table 6: Example of possible error states in translation and back translation.

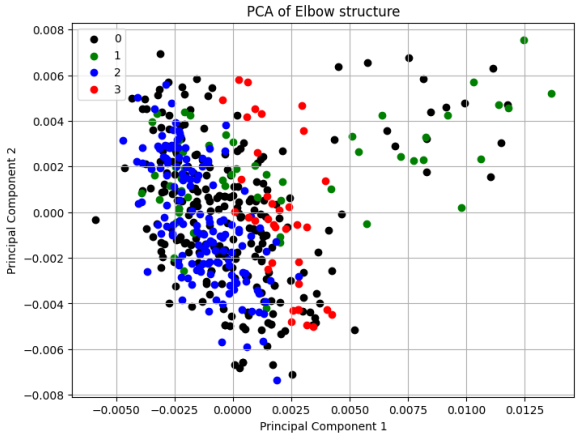


Figure 6: Data distribution over two main principle components