Aligning facts stored in Wikidata to sentences present in Indic Wikipedia

Team members: Vijay Vardhan Alluri, Meghana Bommadi, Shivprasad Sagare, Swayatta Daw

Project Mentor: Tushar Abhishek

Introduction:

Wikidata is a free and open knowledge base that can be read and edited by both humans and machines. It is a central storage repository containing structured data in the form of triples. However, most of the facts in Wikidata are stored in English language and there is no direct way of linking these facts to sentences present in Wikipedia articles written in other languages. Alignment of these facts (in English) to sentences in different languages will help in obtaining more accurate language labels for Wikidata entries making Wikidata entries more multi-lingual.

We, as part of this project, intend to align the facts stored as triples in Wikidata to sentences in Hindi Wikipedia in order to enhance Wikidata in Hindi.

Dataset:

English Wikidata triples and Hindi Wikipedia articles from the multiple domains extracted using “petscan” and wikidata querying.

Papers referred and summaries:

1. Aligning Texts and Knowledge Bases with Semantic Sentence Simplification
2. A Glimpse into Babel:An Analysis of Multilinguality in Wikidata
3. Aligning Knowledge and Text Embeddings by Entity Descriptions
4. T-REx: A Large-Scale Alignment of Natural Language with Knowledge Base Triples
5. Co-training Embeddings of Knowledge Graphs and Entity Descriptions for Cross-lingual Entity Alignment
6. A Survey of Cross-lingual word embedding models

1. Aligning Texts and Knowledge Bases with Semantic Sentence Simplification

Finding the natural language equivalent of structured data is both a challenging and promising task. The authors present an approach to build a dataset of triples aligned with equivalent natural language sentences which would benefit many applications including natural language generation, information retrieval and text simplification.

The approach consists of a 3 step process to reach the end goal of creating a dataset of aligned tripes and sentences.

- Automatically annotating the target textual corpus by linking textual mentions to knowledge base concepts and instances and extracting triples from KB corresponding to entities/relations found in sentences. In our case, this needs to be done in a cross-lingual setting as triple data and sentences are in different languages.
- Crowdsourcing the task of semantic simplification in order to obtain the final dataset of KB facts aligned with natural language sentences from the initial automatically annotated corpus. The task of sentence simplification is as follows: Given a sentence, a set of textual mentions linked to a set of KB instances and concepts and a set of triples, shorten the sentence S as much as possible according to a few rules such as
    - Keep the textual mentions referring to the subject and object of candidate facts.
    - Keep the relations expressed between textual mentions in the sentence.
    - Keep the order of words from the original sentence as much as possible.
    - Ensure that the simplified sentence is grammatical and meaningful.
    - Avoid using external words to the extent possible.
- Final step is to use simplifications as alignments.

Relevance to the project: We had referred this paper to get an idea of related previous work done for creating alignments between Knowledge Base facts and natural language text. We had found this insightful but this approach does not entirely work for us as we are working with Hindi Wikipedia. A few challenges are annotating the target sentences with entities and relations in a low-resource language like Hindi, crowdsourcing and a lot of manual effort etc. Overall, the pipeline to build alignment mapping seems promising but we will need to change some steps for adapting to challenges as stated above.

The dataset created by the authors is available at link

2. A Glimpse into Babel: An Analysis of Multilinguality in Wikidata

Wikidata labels are the way humans interact with the data. In this paper, the authors had explored the state of languages in Wikidata, especially in regard to its ontology and the relationship to Wikipedia. The authors tried to answer three major questions i.e.

- What is the state of Wikidata with regard to multilinguality?
- Is there a difference in the multilinguality of the ontology, compared to the overall multilinguality of the knowledge base?
- How does Wikidata's label distribution relate to the real world and Wikipedia's language distribution?

In order to find the state of languages in Wikidata, the methods they followed are:

- Looking at entity labels, analysing a database dump of Wikidata to count all labels that are noted to be in a certain language. They had excluded redundant information and calculated percentages for each language.
- To understand the language distribution of the ontology, they assessed the property labels via a SPARQL query. To understand how diverse the language distribution of Wiki-data is, we compared it with the native speakers in the world and Wikipedia.
- The distribution of native language speakers is compared with the labels in Wikidata to see how well the language communities are covered by human-readable knowledge in Wikidata. They also used the ranking of Wikipedias based on the numbers of articles for each language version. They then compared this to the ranking by label count, to see whether they can find a similar pattern of languages and get an insight into the relation between Wikipedia's and Wikidata's multilingual information.

They concluded by saying there is still much room for improvement on the current state; as with most of the web, Wikidata's knowledge is mostly available in a few languages.

Relevance to the project: This paper did not really help us with an approach to solve our problem but gave us an insight into the Wikidata and how diverse it is. Most languages, including Hindi, have close to no coverage even though they are spoken by large parts of the world.

3. Aligning Knowledge and Text Embeddings by Entity Descriptions

The aim of this paper is to jointly embed Knowledge Base entities and text corpus. The key issue is to make the alignment model while making sure the vectors of entities, relations and words are in the same space. The authors had proposed a new alignment model based on text descriptions of entities, without dependency on anchors. The resulting embedding vector of an entity not only fits the structured constraints in KBs but also is equal to the embedding vector computed from the text description.
The dataset used for the text corpus was WIkipedia articles, and the corresponding entity for the article was present in FreeBase.

The approach consists of three steps:

- **Create Knowledge Base embeddings** – The entities are embedded in a vector space. Algorithm used was TransE. A fact triplet is in the form of (h,r,t). Here, h and t are two entities. And r is the relation between them. (Basically, in this space, vector (t) = vector(h) + vector(r). So, the relation is effectively captured in this space.)
- **Create Text Embeddings - ** They used skip-gram to create word embeddings from the corresponding text corpus.
- **Knowledge and Text jointly embedding -** So, in order to train the alignment model, they use entities(Freebase) and their corresponding text description (Wikipedia articles) . Most entities do have a text description. The model tries to learn embeddings for each entity, relation and word (in the corresponding article).

To conclude, the authors had proposed a new alignment model based on entity descriptions for jointly embedding a knowledge base to text corpus.

Relevance to the project: We may use an approach related to this as after we have worked on our baseline methods in the future. The idea is to replace the english text model to a multilingual model so that we can embed the Hindi text corpus in a multilingual space so that it can be aligned with the English Knowledge Base entity embeddings. After jointly training Knowledge and text embeddings, an aligned vector space would be created containing both entity and text embeddings.

4. T-REx: A Large-Scale Alignment of Natural Language with Knowledge Base Triples

In this paper they attempted to create a database of alignment of 3.09 million Wikipedia abstracts to 11 million Wikidata triples, covering more than 600 unique Wikidata predicates. They have built a customizable pipeline for the alignment task. The alignment pipeline uses three different automatic alignment techniques.

Pipeline:

Document Reader: reading and tokenizing sentences in Document

Entity Extraction: NER followed by linking them to an entity linker

Date and Time Extraction : getting TIme in docs using Stanford temporal tagger SUtime

Predicate Linking: it links a sequence to its equivalent KB predicate or any of its aliases in the KB.

Coreference Resolution: Resolving pronouns. They mapped a list of possible pronouns to each KB entity using predicates.

Triple Aligners:

1. *NoSub Aligner:* aligner relaxes the distant supervision assumption and assumes that sentences in one paragraph often have the same subject and forms Alignments without having the subject as aligner.
2. AllEnt Aligner: Every pair of entities in a sentence is mapped to their equivalent KB relations. They used coreference resolution to extract all mentions of the main entity of a paragraph
3. SPO Aligner: It aligns triples not only when the subject and object of a triple are mentioned in a sentence but also when the predicate of the triples has been extracted.

Document Writer: For exporting annotated documents.

Used crowdsourcing for evaluation of T-rex and got an accuracy of 97.8% over the evaluated subset of the dataset.

Relevance to the project: Our project is aligning data by seeing how related they are in wikidata and wikipedia but both are in different languages. This paper also concentrates on alignment but both the document and triple are in the same language. This paper gives us an idea about the methods of alignment and

comparison of methods. It also suggests a pipeline mechanism for this process which can be modified by adding a translation step and can be used to create alignment of facts in wikidata to wikipedia articles in hindi. But we decided not to include translation because translation part removes the whole idea of indic language inclusion and models in indian language and also because of the translation loss.

5. Co-training Embeddings of Knowledge Graphs and Entity Descriptions for Cross-lingual Entity Alignment :

Unlike the 3rd paper, this one explores the embedding of knowledge graph and entity description in a **multi-lingual** space. Their approach performs co-training of two models - the multilingual KG embedding and the description embedding model.

Multi-lingual KG Embeddings :

First, a knowledge model is learnt to preserve entities and relations of each language in a separated embedding space . On top of that, an alignment model jointly captures cross-lingual inferences across language-specific embedding spaces.

Multilingual entity description Embeddings :

Multilingual word embeddings are created from the words present in the corresponding description. Each entity description is converted to a sequence of vectors in the mutli-lingual space.

Co-training :

Both the models are co-trained to jointly align them in the same vector space.

Relevance to the project:

We may use this approach in the future after we have covered our baselines and the other approaches. We can have multilingual knowledge base entity embeddings and multilingual text corpus word embeddings. We can co-train the models to jointly align the KB entity and text embeddings in the same vector space. So, the entity and the corresponding text embeddings will lie in a shared multilingual vector space. This is another possible approach we may explore in the future.

6. A Survey of Cross-lingual word embedding models

Considering the cross-lingual nature of our problem statement, we explored the available methods to create and use multilingual embeddings which project the words from multiple languages in the same vector space. Above blog-post which also has a long paper linked to it gives an overview of existing multilingual word embedding models. With multi-lingual technique, embeddings for every language exist in the same vector space, and maintain the property that words with similar meanings (regardless of language) are close together in vector space.

It mentioned strategies to create multilingual embedding models like

- Monolingual mapping
- Pseudo-cross lingual
- Cross-lingual training
- Joint optimization

For training the word embedding models using above strategies, different input features can be used like

- Word-aligned data: A parallel corpus with word alignments that is commonly used for machine translation
- Sentence-aligned data: A parallel corpus without word alignments
- Document-aligned data: A corpus containing documents in different languages
- Lexicon: A bilingual or cross-lingual dictionary with pairs of translations between words in different languages.
- No parallel data: No parallel data whatsoever

After going through the above concepts we further explored available methods to create word embedding models like MUSE by Facebook. We also explored readily available one like fasttext multilingual as well as transformer based multilingual models.

Relevance to the project: The problem statement we are dealing with is cross-lingual in nature. Such problems can be solved either by translating one language data into another and using those monolingual embeddings or by using a cross-lingual embedding approach. We are dealing with Hindi sentences and English Wikidata triples, hence using these cross-lingual word embedding models would benefit us in the project. We would be exploring the use of static as well dynamic contextual embeddings using transformer architectures.

Proposed mechanisms:

1. English Word embedding models:

   Translating Hindi sentences into English. Then using the available English word embeddings like word2vec, we can encode both the wikidata triples and translated sentences.

   The alignment can be calculated using the vector similarity of the embeddings.

2. Multilingual Word embedding models:

   Multilingual word embedding models can be used to encode both triples and sentences into one common vector space

   Similar to the previous method the alignment can be found by vector similarity. This can help in eliminating the translation loss.

3. Creating knowledge graph embeddings:

   Using the methods which leverage the knowledge graph structure , we can prepare embeddings for its entities and relationships for each triple in a vector space . Then the indic data can be transformed to multilingual sentence embeddings and then we can align the embeddings and compute the similarity.

4. NER and Word overlap:

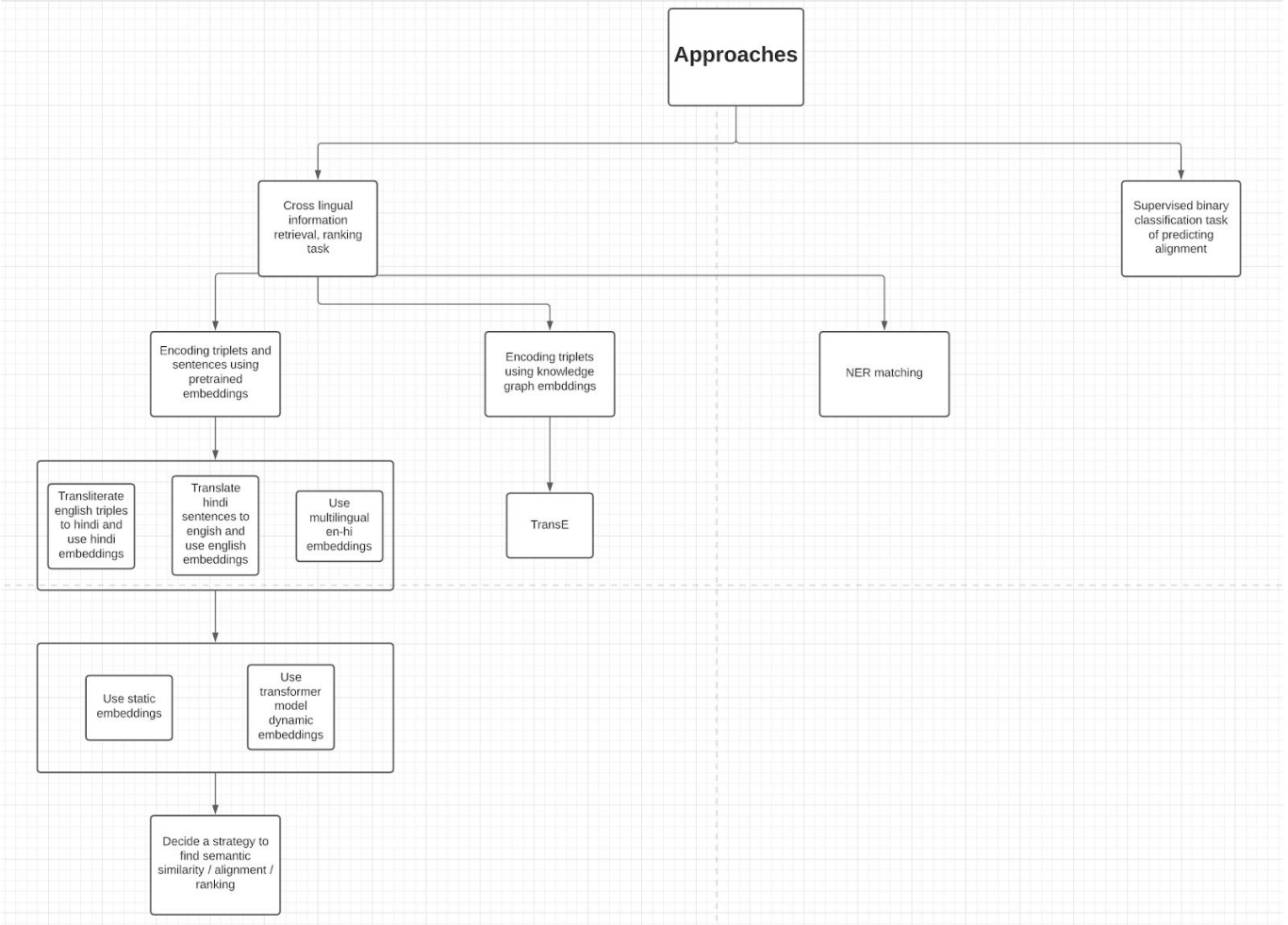
   We extract the subject and object from sentences. Translate the words and then perform word overlap with wikidata triples for finding most similar mappings.

5. Using the corresponding english article:

   Taking english labels in the wikidata Use that english wikipedia article and form multilingual sentence embeddings. Using the relevant hindi wikipedia article and form multilingual sentence embeddings. (this embedding will reduce the translation loss) Use cosine similarity technique to see how many sentences are similar in hindi and english articles

6. Triple formation:

   Extracting subject, object and predicate from hindi sentences and forming triplets. Then encoding them into multilingual embeddings. The same embedding is done for wikidata triplets and finally vector similarity comparison to check the similarity of both. (anaphora will be a huge concern while extracting subject and object)



Baseline for the project:

Basic Steps:

1. Forming multilingual embedding for each triplet in Wikidata.
2. Forming multilingual embeddings for hindi sentences in wikipedia articles.
3. Calculating the similarity of both the embeddings using techniques like cosine similarity.

With this method we can eliminate translation loss, coreference resolution since we are using sentence as a whole and also this is a computationally optimised solution in comparison to the models proposed.

THE END