Fine-Grained Entity Recognition

A B.Tech Project Report Submitted
in Partial Fulfillment of the Requirements
for the Degree of

Bachelor of Technology

by

Khandesh Sailokesh , Jagana vineeth
(180101035 , 180101032)

under the guidance of

Dr.Amit Awekar

to the

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI
GUWAHATI - 781039, ASSAM

CERTIFICATE

*This is to certify that the work contained in this thesis entitled "**Fine-Grained Entity Recognition**" is a bonafide work of **Khandesh Sailokesh , Jagana vineeth** (**Roll No. 180101035 , 180101032**), carried out in the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati under my supervision and that it has not been submitted elsewhere for a degree.*

Supervisor: **Dr.Amit Awekar**

Assistant/Associate Professor,

May, 2024 — Department of Computer Science & Engineering,

Guwahati. — Indian Institute of Technology Guwahati, Assam.

Acknowledgements

Our first experience of Project has been successful, thanks to the department for their support and many of our other teammates with gratitude .We wish to acknowledge all of them. However we wish to make special mention of the following.

First and foremost, we want to express our gratitude to God for allowing us to complete this job successfully. Then we will thank our Supervisor Dr.Amit Awekar(Associate Professor) due Sir's guidance we are able to now know much more practical knowledge on real life scenario's of applications.Should not forget his mentor ship throughout the semester. Mainly he has allotted some M.tech and P.hd students for our convenience, under whose guidance we learned a lot about this project. Their suggestions and directions have helped in the completion of this project.

we'd want to express our gratitude to my parents and friends, who have provided invaluable advice and recommendations during the project's development.

Contents

Chapter 1

Introduction

This project focuses on comparative analysis of data-sets and performance of different models on different data-sets that we have collected on ”Fine-Grained entity recognition” till date.

In this project we have collected various models and different data-sets that are used in various research papers and analysed the recent developments like upgrading to Ultra-Fine entity typing from Fine-Grained entity recognition.

1.1 Problem Definition

The study on various data-sets used in ”Fine-Grained entity recognition” topic and getting non-trivial observations out of data-sets statistics. Studying the various models performances on various data-sets.

1.2 Description

Identifying entity mentions (such as Barack Obama, president, or he) in natural language text and classifying them into preset categories like a person, location, or organisation is the standard entity type tagging task. Type tagging is beneficial for a range of natural language activities, such as co-reference resolution and relation extraction, as well as downstream processing, such as question answering.

Type tagging models use specialised types of data-sets for training, testing and development, which have few characteristics like labeling methods, number of entities etc..(data-sets statistics) based on which performance of the models is varied drastically. Few popular data-sets include FIGER, OntoNotes.

Further we have listed some open source papers for our analysis. We have touched on the functionalities of the data-sets, also persistence and uniformity of data-sets is visualized.

1.3 Challenges and Motivation

From the papers we have collected, not all papers have the code implementation and we have tried to implement the models which have code available on GitHub in that we have successfully ran four models and we faced problems like version shifting i.e., codes have older version implementation that we were not able to download the versions, So we need to convert the whole code to recent version of language for implementation.

Few data-sets were not available in the open source for downloading (Ex: WiFiNE) and also less was known about few data-sets for further research.

The motivation is to understand the evolution of data-sets in this topic and observing the improvement in the performance of models over the period of time and compiling all data-sets used in this topic at one place so that it is useful for further research in this topic.

Chapter 2

Review of Prior Works

We have collected 24 research papers on this topic and compiled all the formal details like name of researcher, number of citations, name of publisher and year of publication.This has helped us in further progress in the project.

For greater clarity in the process, we collected the names of all datasets (In total we have got 20 Datasets(Training ,testing and development)) used in each paper in a table and got all the code of the each individual paper(if the code is available on the net). We have cloned all the models that are available on our local computer for compilation.

2.1 Conclusion

Before starting the acutual study on Datasets and Models collecting all the research papers available and compliing it at on sheet is curcial in upcoming chapter we listed out all collected datasets and models.

Chapter 3

Data-set Characteristics

Every Data-set has specific characteristics through which we can compare efficiency of a given data-set with the other data-set. For comparing any two data-sets we use the characteristics like size of tag set used, number of entity mentions, number of sentences, number of labels, number of tokens and labelling method. So, to understand these we go through the definitions.

3.1 Definitions

3.1.1 Tag set

The collection of word classes/tags used for particular task here for Fine-Grained entity tagging is called Tag set.

person	doctor
actor	engineer
architect	monarch
artist	musician
athlete	politician
author	religious_leader
coach	soldier
director	terrorist

organization	terrorist_organization
airline	government_agency
company	government
educational_institution	political_party
fraternity_sorority	educational_department
sports_league	military
sports_team	news_agency

location	body_of_water
city	island
country	mountain
county	glacier
province	astral_body
railway	cemetery
road	park
bridge	

product	camera
engine	mobile_phone
airplane	computer
car	software
ship	game
spacecraft	instrument
train	weapon

art	written_work
film	newspaper
play	music

event	military_conflict
attack	natural_disaster
election	sports_event
protest	terrorist_attack

building	time	chemical_thing	website
airport	color	biological_thing	broadcast_network
dam	award	medical_treatment	broadcast_program
hospital	educational_degree	disease	tv_channel
hotel	title	symptom	currency
library	law	drug	stock_exchange
power_station	ethnicity	body_part	algorithm
restaurant	language	living_thing	programming_language
sports_facility	religion	animal	transit_system
theater	god	food	transit_line

Fig 1 : This is the tag set of the FIGER model which has 112 tags.

3.1.2 Entity

An entity can be any word or series of words that consistently refers to the same thing. Every detected entity is classified into a predetermined category.

Entity extraction is a text analysis technique that uses Natural Language Processing (NLP) to automatically pull out specific data from unstructured text, and classifies it according to predefined categories. These categories are named entities, the words or phrases that represent a noun.

3.1.3 Sentence

A Sentence contains entity mentions and the goal in fine grained entity typing is to classify the entity mentions in every sentence of the dataset and classify it into free-form phases i.e., eg: Singer, Animals, City, Games. Collection of sentences forms a Dataset

3.1.4 Label

Every entity mentions in a sentence is given a specific identity(hierarchical identity) from the predefined tag set is called label of a given entity mention.

Labeling method

Every dataset used(training, testing and development)for this particular research is either manually labelled dataset or automatically labelled dataset.

For example FIGER is automatically labelled dataset whereas FIGER-GOLD(training dataset) is manullay labelled dataset.

3.1.5 Token

"Tokens" are usually single words (at least in languages like English), and "tokenization" is the process of breaking down a text or set of text into individual words. These tokens are then utilised as input for various analyses and operations.

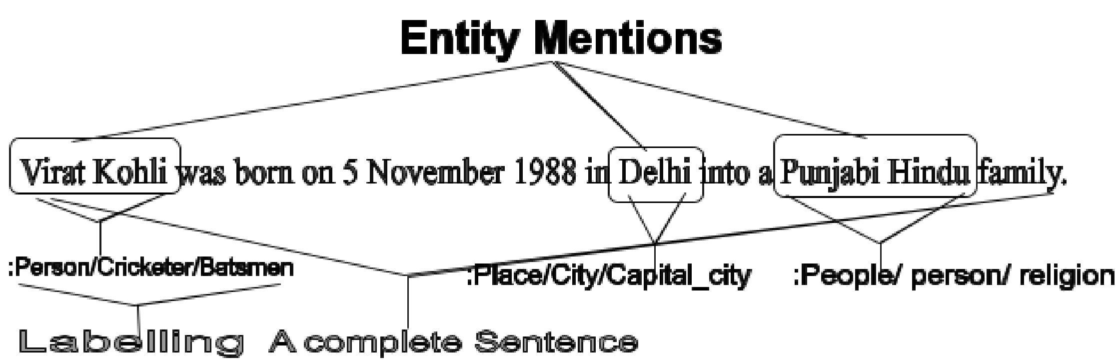


Fig 2 : This example helps us to understand the above definitions better

3.2 Datasets

After Going through the 24 research papers we have collected ,we came across 19 Datasets used by the researcher in this topic. Here is the list of all the Datasets we collected.

- FIGER
- FIGER(GOLD)
- Stanford(CoNLL)
- NEL
- Wiki
- BBN
- OntoNotes
- Open Entity
- OntoNotes5.0
- WiFiNE
- Wiki-FbF
- Wiki-FbT
- 1k-WFB-g
- DBpedia
- WIKI-AUTO
- WIKI-MAN
- AIDA
- FEW-NERD
- YAGO

3.3 Models

We have got the following models from the research papers and we here by list out the model names for better comparison between the models on the given data-sets.

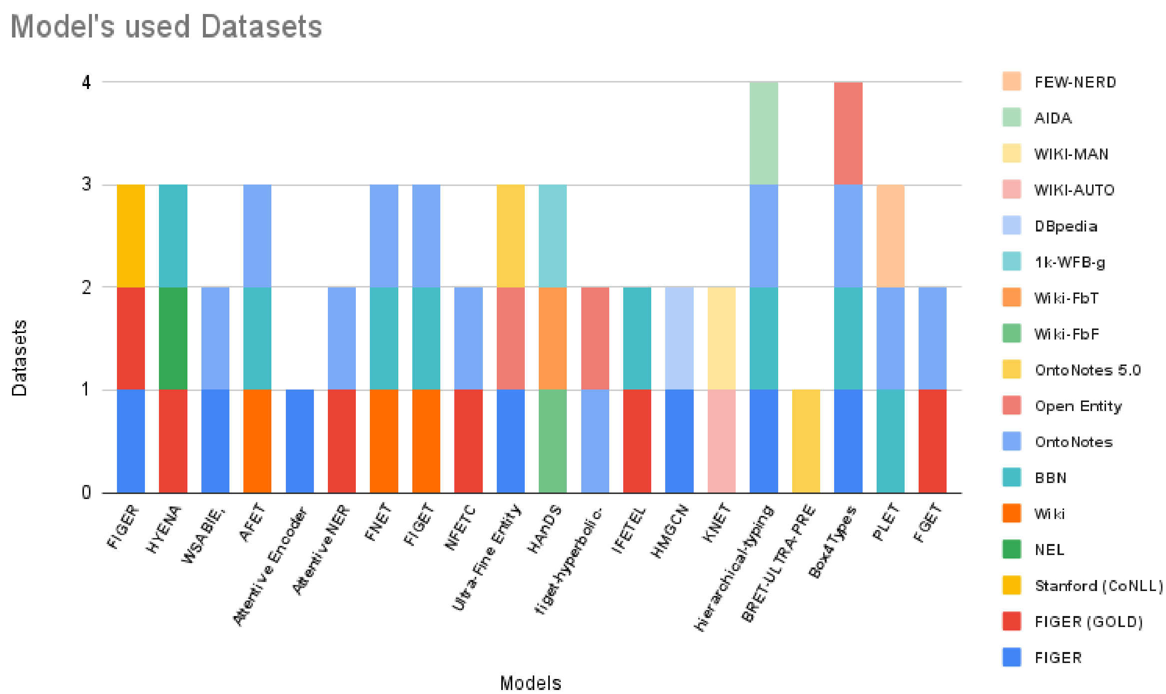
1. FIGER [LW12]
2. HYENA [YBH+12]
3. [GLG+14]
4. AFET [RHQ+16]

5. WSABIE [YGL15]
6. [SSIR16a]
7. NFGEC [SSIR16b]
8. FNET [AAA17]
9. FIGET [ZDVD18]
10. NFETC [XB18]
11. Ultra-Fine Entity typing [CLCZ18]
12. [GL18]
13. figet-hyperbolic-space [LHS19]
14. IFETEL [DDLS19]
15. HMGCN [JHLD19]
16. [ATM+19]
17. KNET [XLLS18]
18. hierarchical-typing [CCVD20]
19. [BD21]
20. MLMET [DSW21]
21. Box4Types [OBMD21]
22. PLET [DCH+21]
23. FGET [HWZ21]

In the above list of models we consider the numbering of the models as a reference in our further comparison between the models on a given dataset.

3.4 Model's Datasets

In each research paper the model is tested on specific datasets, So we here by want to list which model uses which dataset through graphical format.On X-axis we listed all models and On Y-axis number of datasets used by each model using bar graph.



Chapter 4

Models Comparison

After going through the models and their results on different datasets, We got to know Different models have different performances on various datasets, So it's a bit tricky to compare any two models. So, we here by want to organize the comparison between the models.

4.1 Comparison criteria

We use the following criteria for comparing the models.

- The comparison between any two Models is done on a given dataset.
- If Model A is Better than Model B on Dataset D, that imply Model A has Better overall performance atleast two better out of precision, recall and F1 Score than Model B on Dataset D.

4.2 Comparison Graphs

The comparison between the models is depicted in the graphical format. The description of the graph is as follows.

- We used Directed graph. Every graph is the comparison between models performance on a given dataset.
- Graph Contains nodes and Directing edges between nodes. Every node represent the model number referencing to the previous chapter as mentioned.
- If there is a directing edge from node 1 to node 2 that mean model representing node 1 is better performing than model representing node 2.

4.2.1 FIGER

Below is the comparison graph of models performance on FIGER dataset.

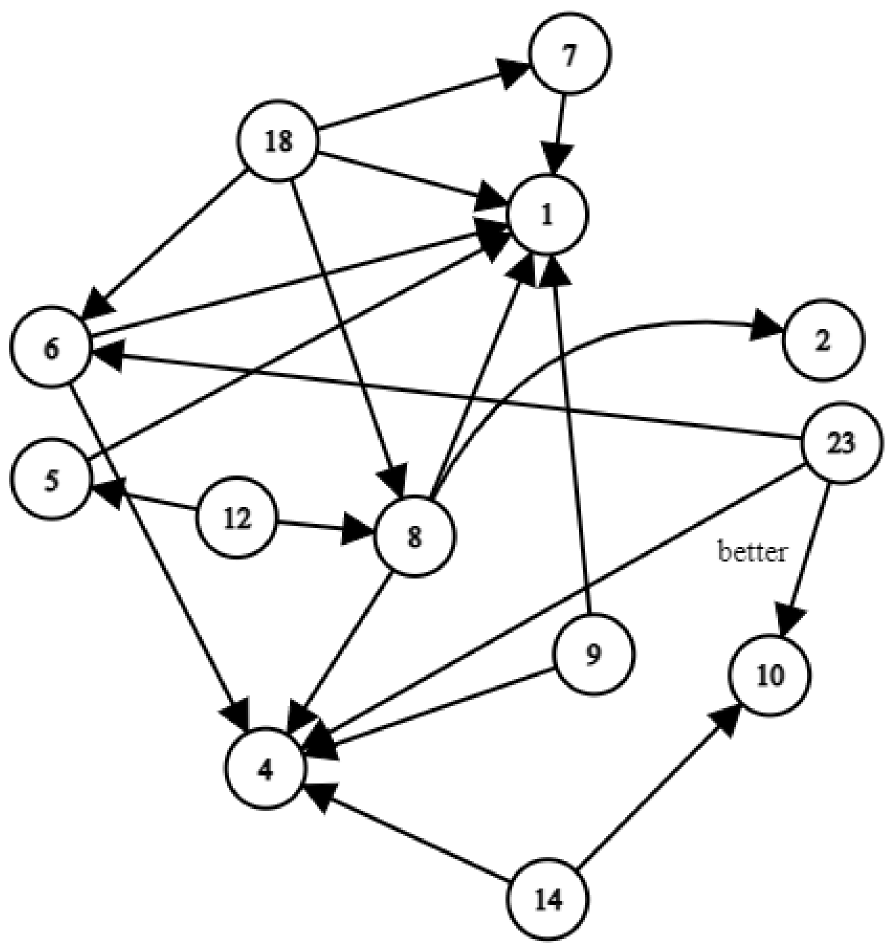


Fig 3

4.2.2 OntoNotes

Below is the comparison graph of models performance on OntoNotes dataset.

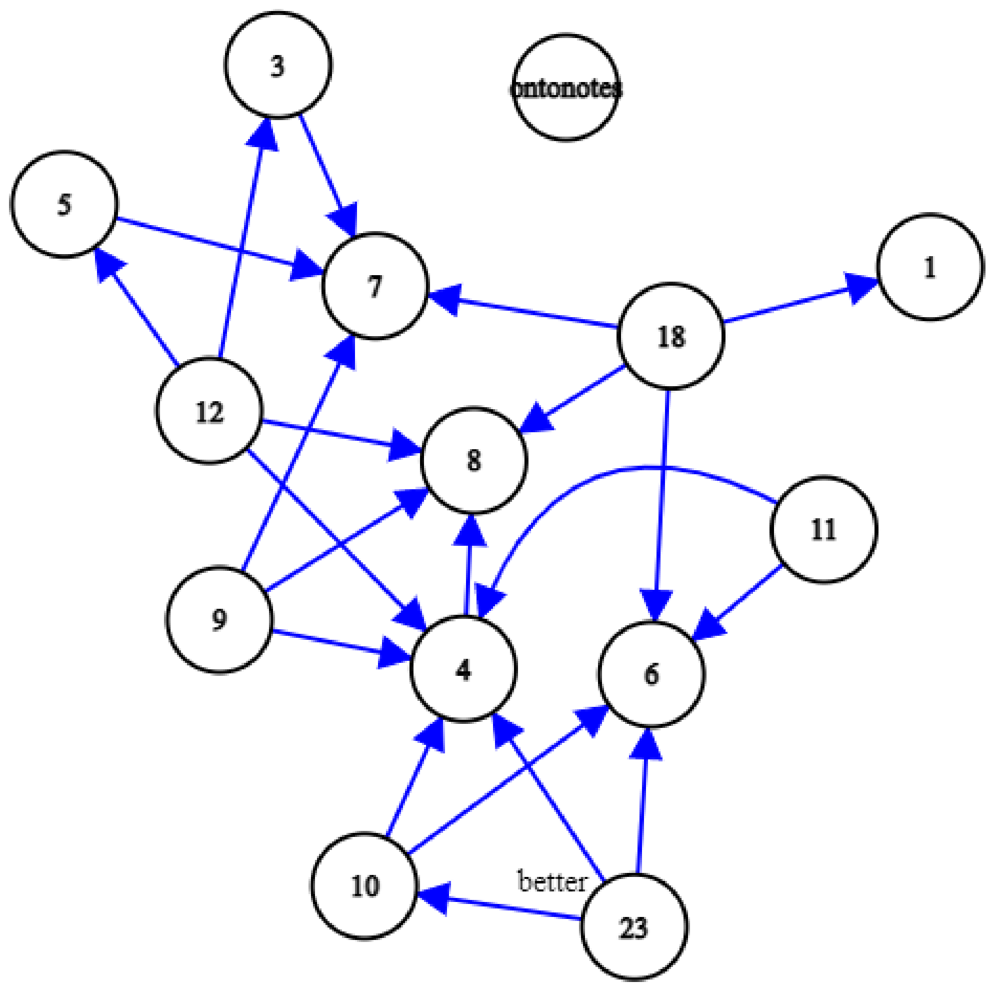


Fig 4

4.2.3 Wiki

Below is the comparison graph of models performance on Wiki dataset.

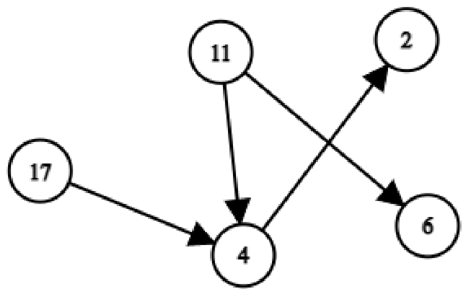


Fig 5

4.2.4 BBN

Below is the comparison graph of models performance on BBN dataset.

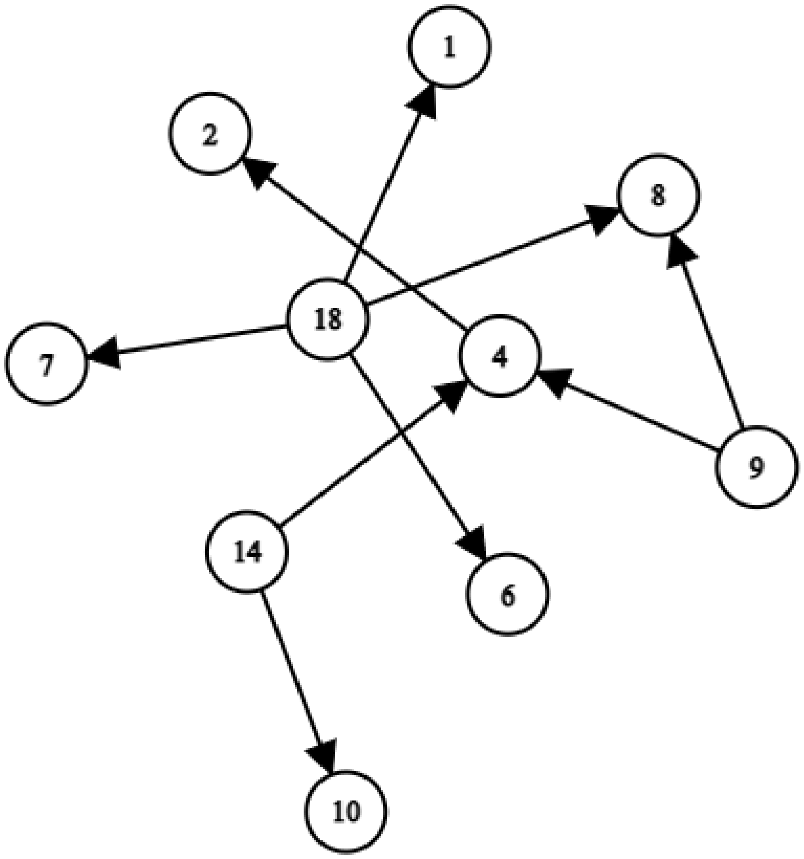


Fig 6

4.2.5 Few Conclusions from Drawing the comparison graphs

- The above comparison graphs formed are *Directed acyclic graphs*. The proof for the same is as follows.

  Let us prove it through contradiction, Let us consider a comparison Cyclic graph possible as shown in the figure.

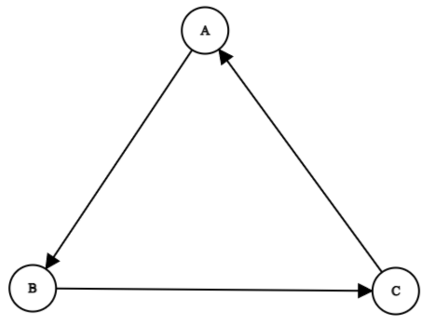


Fig 7

From the graph we can say A > B , B > C and C > A(This is not at all possible as A > C from A > B and B > C).

Hence by contradiction we can say Model comparison graphs formed are *Directed acyclic graphs*.

- As the graphs formed are *Directed acyclic graphs* (DAG) we can apply topological sort Algorithm on a DAG.

- From the topologically sorted graph we can conclude two things on a given dataset we will be able to know the best performing Model and the least performing model.

  So from the above observations and from Fig 3, Fig 4, Fig 5, Fig6 we can conclude:

– From Fig 3 Model Number 23 (FGET: Transfer learning for fine-grained entity typing) is the best performing model and Model Number 1 (FIGER: Fine grained entity recognition) is the least performing model on FIGER dataset.

– From Fig 4 Model Number 23 (FGET: Transfer learning for fine-grained entity typing) is the best performing model and Model Number 6,7 An Attentive Neural Architecture for Fine-grained Entity Type Classification and NFGEC: Neural Architectures for Fine-grained Entity Type Classification) are the least performing models on OntoNotes dataset.

- From Fig 5 Model Number 17 (KNET: Improving Neural Fine-Grained Entity Typing with Knowledge Attention) is the best performing model and Model Number 2 (HYENA : Hierarchical type classification for entitynames.) is the least performing model on Wiki dataset.
- From Fig 6 Model Number 18 (hierarchical-typing: Hierarchical Entity Typing via Multi-level Learning to Rank) is the best performing model and Model Number 1,2 (FIGER: Fine grained entity recognition and HYENA : Hierarchical type classification for entitynames.) are the least performing models on BBN dataset.

4.3 Conclusion

The above results we got are using few criteria for comparing the models but there are few exceptions while concluding the best performing and least performing models on a given dataset as there we don't have comparison between few models.

Chapter 5

Observations

After organising all of the datasets collected and tabulating the dataset statistics, we came across a few observations that we represented in both tabular and graphical format. Here are our findings.

- Among all the datasets that we have collected and counted the number of times each dataset has been used, the most popular dataset, OntoNotes(11 times), is the most frequently used dataset, and the second most frequently used dataset is BBN (8 times).We observe the same using the below bar graph Fig 8.

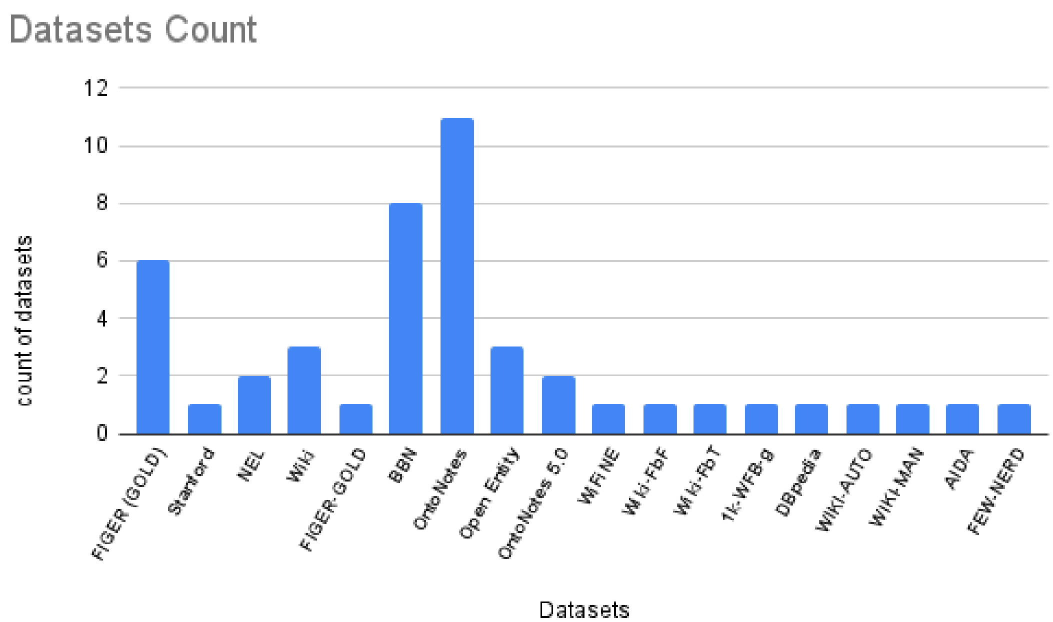


Fig 8

- There is no uniform dataset that exist for fine grianed entity recognition task. Most of the model uses distant supervision. There is no such labeled dataset which can be used for training all the models. We can see the non-uniform data quality using following example that we took from FIGER dataset.

  Eg. training sample:

  Fidel Castro and Che Guevara depart from Tuxpan , Veracruz , Mexico , enroute to Santiago de Cuba aboard the yacht Granma with 82 men .

  entity: Fidel Castro , labels: ['/person/athlete', '/person/actor', '/person/politician', '/person', '/person/soldier', '/person/author']

  entity: Che Guevara , labels: ['/person', '/organization/company', '/person/actor', '/person/author', '/person/doctor', '/person/artist', '/person/politician', '/person/soldier']

  entity: Tuxpan , Veracruz , labels: ['/location/city', '/location']

  entity: Mexico , labels: ['/location', '/person/artist', '/language', '/location/country'] entity: Santiago de Cuba , labels: ['/location', '/location/city']

entity: Granma , labels: ['/building', '/product/ship', '/location']

In the above example Even "Mexico" is referred to as a "language" and a "artist," which is absurd.

- Referencing to the previous point we can say that, The distant supervision technique that was used to automatically build the training corpus assigned multiple labels to the hyperlinks in Wikipedia using their Freebases tags and then mapped those to FIGER types. The main issue is that all of those types could be correct in different scenarios; unfortunately, getting the correct label depends on the local context, which is still the subject of many papers.

- Types(tag) and size of tag set of every model is also different. Like FIGER model is using 112 tag set size, HYENA is using 505 types as tag set size.

- Type coverage Problem: As observed from the available dataset top 5 types covers the 70-80 % of dataset.

  This makes some of the type labels very rare in the data which will have a poor effect on the model.

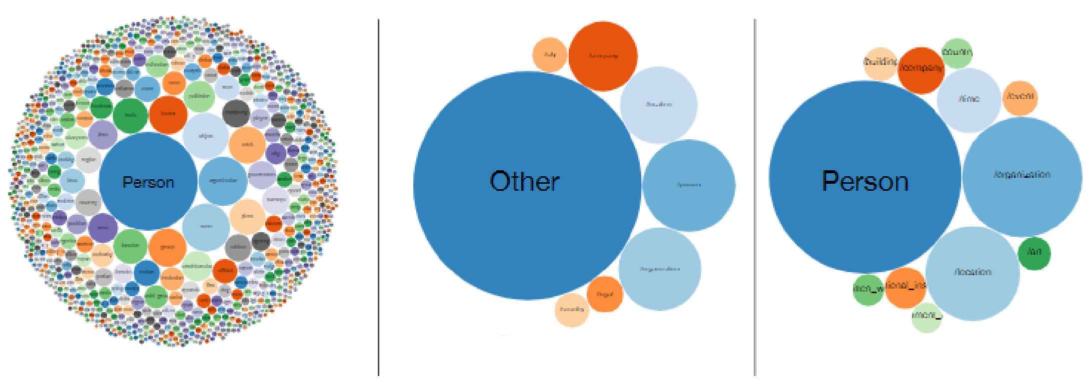


Fig 9: From left to right these are Bubble representation of ultra Fine grained , OntoNotes and FIGER Datasets tag sets labels, where a Bubble's size is proportional to the labels frequency.

- From the fig 9 we can also observe that over the year there is a clear shift from Fine grained data to Ultra Fine grained data, Because Ultra Fine grained data is much more diverse and fine grained when compared to existing dataset(OntoNotes and FIGER).

- The depth of the hierarchy used by different models is different.Number of hierarchy level labelling is also different for different datasets as shown below.

Dataset	number of levels
FIGER	2
FIGER (GOLD)	2
Wiki	2
BBN	2
OntoNotes	3
WIKI-AUTO	2
WIKI-MAN	2
AIDA	3

Fig 10

- Labelling Method of the dataset can also effect the accuracy of the model. A dataset can be Manually labeled or Automated labeled. Below Table gives us an idea about labelling method used for different datasets.

Dataset	labeling method
FIGER	automated
FIGER (GOLD)	manually
Stanford (CoNLL)	manually
Wiki	Automated
BBN	manually
OntoNotes	manually
Open Entity	Automated
Wiki-FbF	Automated(HAnDS framework)
Wiki-FbT	Automated(HAnDS framework)
1k-WFB-g	manually annotated
WIKI-AUTO	Automated
WIKI-MAN	manually(only used for testing)
FEW-NERD	manually

Fig 11

- There is a steep rise in the number of tags/types i.e is tagset size used by the datasets for labelling their over the period of time.Through this as well we can conclude that there is focus shift from fine grianed data to ultra fine grained data. This is clear from below graph.

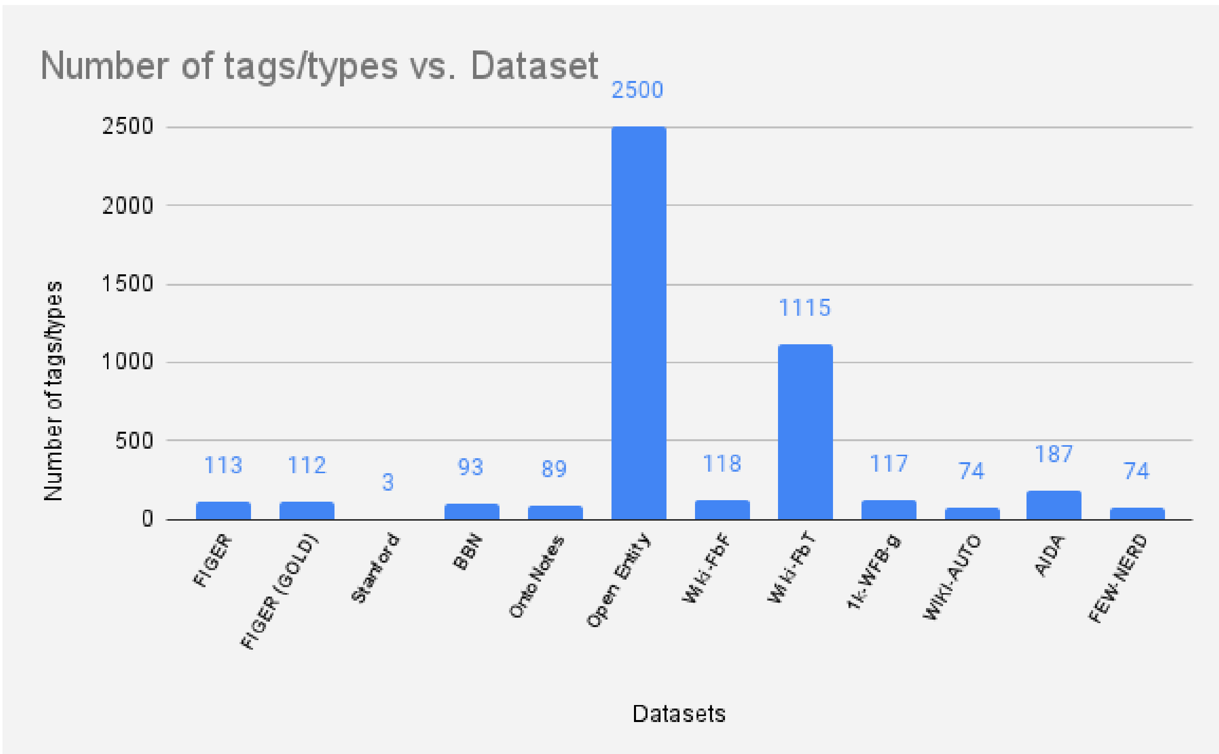


Fig 12

- Another Observation from the dataset statistics that are available is, there is steep increase in the number of sentences in the datasets over the period of time. As number of Entity mentions is directly propositional to number of sentences in a dataset we can also observe a steep increase in the number of Entity mentions in the datasets. Below two graphs depicts the same.

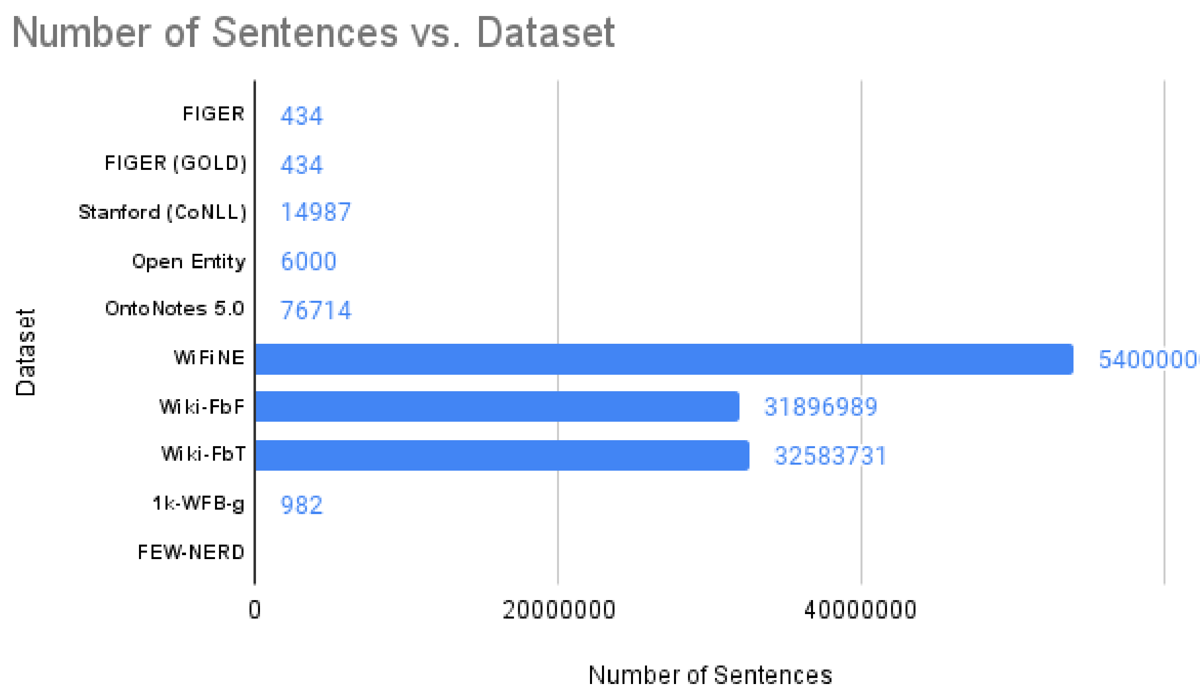

Number of Sentences vs. Dataset
FIGER
FIGER (GOLD)
Stanford (CoNLL)
Open Entity
OntoNotes 5.0
WiFiNE
Wiki-FbF
Wiki-FbT
1k-WFB-g
FEW-NERD
434
434
14987
6000
76714
31896989
32583731
982
Dataset
0
20000000
40000000
Number of Sentences

Fig 13

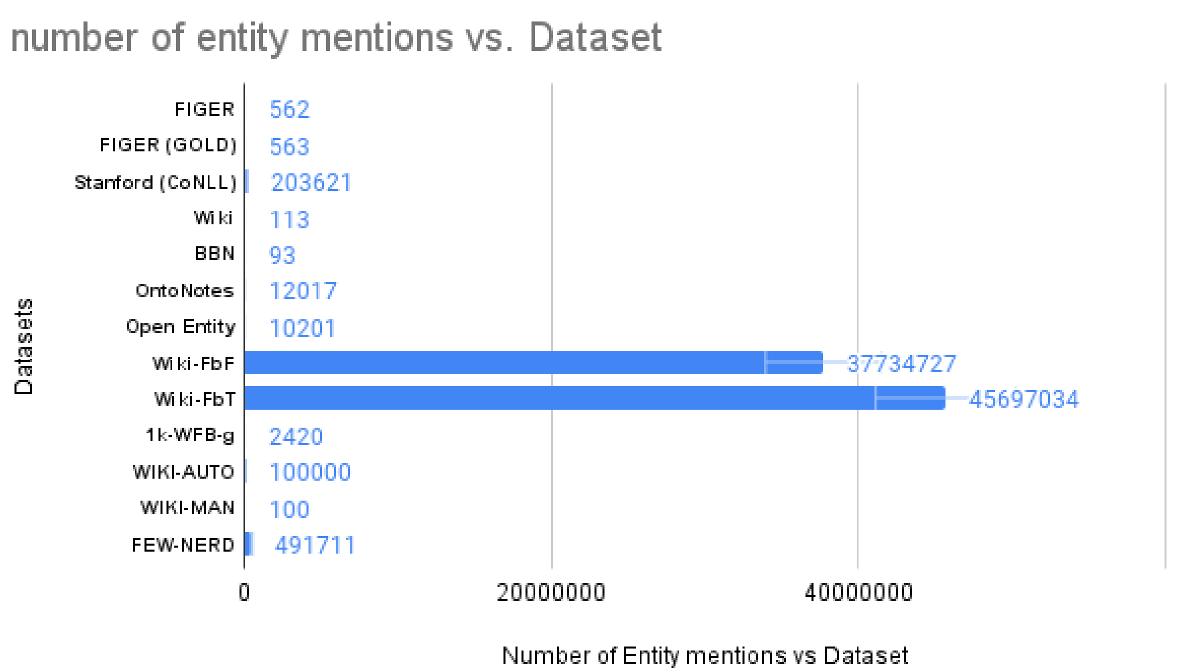

number of entity mentions vs. Dataset
FIGER
FIGER (GOLD)
Stanford (CoNLL)
Wiki
BBN
OntoNotes
Open Entity
Wiki-FbF
Wiki-FbT
1k-WFB-g
WIKI-AUTO
WIKI-MAN
FEW-NERD
562
563
203621
113
93
12017
10201
37734727
45697034
2420
100000
100
491711
Datasets
0
20000000
40000000
Number of Entity mentions vs Dataset

Fig 14

- From Fig 13 and Fig 14 We clearly see that the two datasets WiKi-FbF and WiKi-FbT built in [ATM$^+$19], together with the evaluation resource, are currently the largest available training and testing dataset for the entity recognition problem. They are supported by empirical experimentation to ensure the quality of the built corpora.

Chapter 6

Conclusion

- Upon concluding, As described in the problem definition we have studied the important characteristics about datasets and understood various datasets comparison parameters. We have also plotted various graphs for comparing datasets(i.e dataset vs tag set size, dataset vs sentences,dataset vs entity mentions ,etc). This information about all datasets at one place would definitely help researchers for further research in this topic.It would help researcher to decide which dataset to choose for their research and he can get a good idea of which dataset is better to use for which model etc.

- As mentioned in the problem definition we have also compared the different models efficiency on different datasets through Directed Acyclic graphs format and got which model is the best performing on a given dataset.This information would help those who use these models as part of their work as they get an idea of which dataset to use for their model.We have almost collected 24 research papers and 19 datasets on this research space it will become easier for researcher to go through models and datasets and their efficiencies at one place.

References

[AAA17] Abhishek Abhishek, Ashish Anand, and Amit Awekar. Fine-grained entity type classification by jointly learning representations and label embeddings. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, volume 1, pages 797–807, 2017.

[ATM+19] Abhishek Abhishek, Sanya Bathla Taneja, Garima Malik, Ashish Anand, and Amit Awekar. Fine-grained entity recognition with reduced false negatives and large type coverage. *arXiv preprint arXiv:1904.13178*, 2019.

[BD21] Emanuela Boros and Antoine Doucet. Transformer-based methods for recognizing ultra fine-grained entities (rufes). *arXiv preprint arXiv:2104.06048*, 2021.

[CCVD20] Tongfei Chen, Yunmo Chen, and Benjamin Van Durme. Hierarchical entity typing via multi-level learning to rank. *arXiv preprint arXiv:2004.02286*, 2020.

[CLCZ18] Eunsol Choi, Omer Levy, Yejin Choi, and Luke Zettlemoyer. Ultra-fine entity typing. *arXiv preprint arXiv:1807.04905*, 2018.

[DCH+21] Ning Ding, Yulin Chen, Xu Han, Guangwei Xu, Pengjun Xie, Hai-Tao Zheng, Zhiyuan Liu, Juanzi Li, and Hong-Gee Kim. Prompt-learning for fine-grained entity typing. *arXiv preprint arXiv:2108.10604*, 2021.

[DDLS19] Hongliang Dai, Donghong Du, Xin Li, and Yangqiu Song. Improving fine-grained entity typing with entity linking. *arXiv preprint arXiv:1909.12079*, 2019.

[DSW21] Hongliang Dai, Yangqiu Song, and Haixun Wang. Ultra-fine entity typing with weak supervision from a masked language model. *arXiv preprint arXiv:2106.04098*, 2021.

[GL18] Abbas Ghaddar and Philippe Langlais. Transforming wikipedia into a large-scale fine-grained entity type corpus. In *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*, 2018.

[GLG+14] Dan Gillick, Nevena Lazic, Kuzman Ganchev, Jesse Kirchner, and David Huynh. Context-dependent fine-grained entity type tagging. *arXiv preprint arXiv:1412.1820*, 2014.

[HWZ21] Feng Hou, Ruili Wang, and Yi Zhou. Transfer learning for fine-grained entity typing. *Knowledge and Information Systems*, 63(4):845–866, 2021.

[JHLD19] Hailong Jin, Lei Hou, Juanzi Li, and Tiansi Dong. Fine-grained entity typing via hierarchical multi graph convolutional networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4969–4978, 2019.

[LHS19] Federico López, Benjamin Heinzerling, and Michael Strube. Fine-grained entity typing in hyperbolic space. *arXiv preprint arXiv:1906.02505*, 2019.

[LW12] Xiao Ling and Daniel S Weld. Fine-grained entity recognition. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[OBMD21] Yasumasa Onoe, Michael Boratko, Andrew McCallum, and Greg Durrett. Modeling fine-grained entity types with box embeddings. *arXiv preprint arXiv:2101.00345*, 2021.

[RHQ+16] Xiang Ren, Wenqi He, Meng Qu, Lifu Huang, Heng Ji, and Jiawei Han. Afet: Automatic fine-grained entity typing by hierarchical partial-label embedding. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 1369–1378, 2016.

[SSIR16a] Sonse Shimaoka, Pontus Stenetorp, Kentaro Inui, and Sebastian Riedel. An attentive neural architecture for fine-grained entity type classification. *arXiv preprint arXiv:1604.05525*, 2016.

[SSIR16b] Sonse Shimaoka, Pontus Stenetorp, Kentaro Inui, and Sebastian Riedel. Neural architectures for fine-grained entity type classification. *arXiv preprint arXiv:1606.01341*, 2016.

[XB18] Peng Xu and Denilson Barbosa. Neural fine-grained entity type classification with hierarchy-aware loss. *arXiv preprint arXiv:1803.03378*, 2018.

[XLLS18] Ji Xin, Yankai Lin, Zhiyuan Liu, and Maosong Sun. Improving neural fine-grained entity typing with knowledge attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[YBH+12] Mohamed Amir Yosef, Sandro Bauer, Johannes Hoffart, Marc Spaniol, and Gerhard Weikum. Hyena: Hierarchical type classification for entity names. In *Proceedings of COLING 2012: Posters*, pages 1361–1370, 2012.

[YGL15] Dani Yogatama, Dan Gillick, and Nevena Lazic. Embedding methods for fine grained entity type classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint*

Conference on Natural Language Processing (Volume 2: Short Papers), pages 291–296, 2015.

[ZDVD18] Sheng Zhang, Kevin Duh, and Benjamin Van Durme. Fine-grained entity typing through increased discourse context and adaptive classification thresholds. *arXiv preprint arXiv:1804.08000*, 2018.