Master Degree in Computer Science

Natural Language Processing

Prof. Alfio Ferrara

Dott. Sergio Picascia, Dott. Davide Riva, Dott.ssa Elisabetta Rocchetti, Dott.ssa Darya Shlyk

Department of Computer Science, Università degli Studi di Milano
Room 7012 via Celoria 18, 20133 Milano, Italia alfio.ferrara@unimi.it

Ideas for final projects

Instructions

The final project consists in the preparation of a short study on one of the topics of the course, identifying a precise research question and measurable objectives. The project will propose a methodology for solving the research question and provide an experimental verification of the results obtained according to results evaluation metrics. The emphasis is not on obtaining high performance but rather on the critical discussion of the results obtained in order to understand the potential effectiveness of the proposed methodology.

The results must be documented in a short article of not less than 4 pages and no more than 8, composed according to the guidelines available here: template and using the corresponding $LaTeX$ or MS Word templates. Students have also to provide access to a GitHub repository containing the code and reproducible experimental results.

Finally, the project will be discussed after a 10 minutes presentation in English with slides.

Procedure

Exam dates are just for the registration of the final grade. The project discussion will be set by appointment, according to the following procedure:

1. Subscribe to any available date
2. Contact Prof. Ferrara as soon as
    1. The project is finished and ready to be discussed
    2. After the date of your subscription is expired
3. Setup an appointment and discuss your work

Example: you subscribe the exam date of [Month] [Day]. **Anytime after [Month] [Day]**, when the **project is ready**, you will contact Prof. Ferrara and set an appointment. You discuss the project during the appointment.

If you are **interested in doing your final master thesis on these topics**, the final project may be a preliminary work in view of the thesis. In this case, discuss the contents with Prof. Ferrara.

Structure of the paper

1. **Introduction**

   Provides an overview of the project and a short dicsussion on the pertinent literature

2. **Research question and methodology**

   Provides a clear statement on the goals of the project, an overview of the proposed approach, and a formal definition of the problem

3. **Experimental results**

   Provides an overview of the dataset used for experiments, the metrics used for evaluating performances, and the experimental methodology. Presents experimental results as plots and/or tables

4. **Concluding remarks**

   Provides a critical discussion on the experimental results and some ideas for future work

Project ideas

The following are ideas for projects. For each idea, a short description, example of datasets that can be used, and bibliographic references are provided. Students may **choose one of the following** as their project theme or **they can propose their own idea**, structuring the proposal as those presented in this document. In the latter case, just send the project description to Prof. Ferrara.

Stop it, it's forbidden! (P1)

Commercial Large Language Models (LLMs), such as ChatGPT, CoPilot, Gemini, have become ubiquitous in various applications, from chatbots to content generation. However, concerns persist regarding their ideological biases, potential censorship, and the need for effective safeguards VS the risk of safeguards as tools for censoring information. This project aims to explore one or more of the following aspects, always by proposing a statistical approach for performing large scale tests, measuring the obeserved evidences and provide set of preliminary results to show up the effectiveness of the proposed approach.

- **Ideological Biases:** Investigate whether commercial LLMs exhibit biases related to political, cultural, or social ideologies. Analyze their responses to prompts on sensitive topics and assess any inherent bias.
- **Safeguards Evaluation:** Examine existing safeguards implemented by commercial LLM providers. Evaluate their effectiveness in preventing harmful or biased outputs. Consider transparency, explainability, and adaptability of these safeguards.
- **Censorship Risks:** Assess the risk of inadvertent or intentional censorship by LLMs. Explore scenarios where information is withheld due to political pressure, corporate interests, or other factors.
- **Comparative Analysis:** Compare different commercial LLMs (such as ChatGPT, CoPilot, Gemini, etc.) in terms of biases, safeguards, and censorship risks. Investigate variations across languages, regions, and user queries.

Censorship and idealogical biases can be studied on one or more of this tasks:

- **Text-to-Text generation:** Observe the effect of different prompt inputs and how they may trigger safeguards or ideologically biased answers in the LLM(s). *Some examples: observe how LLMs paraphrase input texts containing "sensitive" words; ask for information about different books or contents with different levels of "ideological risk" and observe the answer of the LLM.*
- **Text-to-Image generation:** Explore the limits of the type and kind of images that are generated according to different prompts introducing different levels of sensistivity in the request. *Some examples: observe the reaction to different requests to generate images containing naked people, violence, unethical contents and so on.*
- **Image-to-Image transformation:** Explore patch generation in image impainting according to the level of sensitivity of the deleted original patch. *Some examples, observe the inpaining of images containing naked people, violence, unethical contents and so on.*
- **Image-to-Text generation:** observe and analyze the text produced to describe images containing contents with different levels of sensitivity. *Some examples, observe the text generated from images containing naked people, violence, unethical contents and so on.*

Dataset

- Any dataset containing sensitive contents as well as non sensitive contents, including automatically generated datasets.

References

- Glukhov, D., Shumailov, I., Gal, Y., Papernot, N., & Papyan, V. (2023). Llm censorship: A machine learning challenge or a computer security problem?. *arXiv preprint arXiv:2307.10719*.
- Deng, Y., & Chen, H. (2023). Divide-and-Conquer Attack: Harnessing the Power of LLM to Bypass the Censorship of Text-to-Image Generation Model. *arXiv preprint arXiv:2312.07130*.
- Urman, A., & Makhortykh, M. (2023). The Silence of the LLMs: Cross-Lingual Analysis of Political Bias and False Information Prevalence in ChatGPT, Google Bard, and Bing Chat.
- Zhou, X., Wang, Q., Wang, X., Tang, H., & Liu, X. (2023). Large Language Model Soft Ideologization via AI-Self-Consciousness. *arXiv preprint arXiv:2309.16167*.

Hurry up, I'm hungry! (P2)

The project aims at developing a system that generates culinary recipes based on a given list of ingredients. The system will use large language models (LLMs) and statistical methods to compose ingredients and cooking methods. The generated recipes will be evaluated by comparing them to the ratings of similar real recipes.

The idea is to exploit a pre-trained language model like GPT-4, BERT or others. This model can be fine-tuned on the recipe dataset to enhance its understanding of culinary contexts. The system will then be capable of generating recipes by combining user-provided ingredients with cooking methods learned during training. To refine these recipes, statistical methods will be employed. By analyzing correlations between ingredients, cooking methods, and recipe ratings, the system can identify combinations that are typically well-received. This analysis will guide the generation process, making it more likely to produce high-quality recipes. The generated recipes will be evaluated against real recipes. This comparison will involve using existing user ratings of similar real recipes as a benchmark. By assessing the generated recipes against these ratings, the system's performance can be objectively measured.

Dataset

Any dataset available online providing recipes with ingredients and cooking methods as well as user ratings.

References

- Ko, H., Lee, S., Park, Y., & Choi, A. (2022). A survey of recommendation systems: recommendation models, techniques, and application fields. *Electronics*, *11*(1), 141.
- Zangerle, E., & Bauer, C. (2022). Evaluating recommender systems: survey and framework. *ACM computing surveys*, *55*(8), 1-38.
- Jabeen, H., Weinz, J., & Lehmann, J. (2020, July). AutoChef: Automated generation of cooking recipes. In *2020 IEEE Congress on Evolutionary Computation (CEC)* (pp. 1-7). IEEE.

News from the past (P3)

> Events are commonly considered as the building blocks of historical knowledge with which historians construct their system of ideas about the past. A systematic and consistent analysis of events mentioned in historical texts would greatly contribute to a better understanding of large archives in this domain. [1]

Event dection in text is a challenging task consisting in finding text portions reporting the description of a real event. Historical events are a specific type of events that are framed in a historical context, reporting facts, dates, historical figures and locations. One of the challenges is that the definition of historical event itself is problematic. Another challenge is due to the fact that, when dealing with corpora of historical documents, the language itself may change dealing to shifts in semantics and style.

The project aims at addressing the task of detecting historical events. In particular, the project will provide:

1. **an operative definition of the notion of historical event**. This definition is part of the project objectives and it is the basis for the event extraction methodology. Focusing on specific types of events, e.g., specific people events, topic-related events, short/long term events, is possible.

2. a **methodology for extracting the required components of the event from text** , according to the event definiton at step 1, e.g., the agent, the event itself, the location, context, etc.
3. an **overview and statistics about the events** extracted from a historical corpus of interest (see below).
4. a **case study reconstructing the main events in a given time period** from the corpus of interest.

Dataset for training and evaluation

HISTO dataset, providing annotation guidelines designed to detect and classify event mentions in texts and a corpus of historical texts annotated with events (span + class). link

References

- Atefeh, F., & Khreich, W. (2015). A survey of techniques for event detection in twitter. Computational Intelligence, 31(1), 132-164. link
- Shan, D., Zhao, W. X., Chen, R., Shu, B., Wang, Z., Yao, J., ... & Li, X. (2012, August). Eventsearch: a system for event discovery and retrieval on multi-type historical data. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1564-1567). link
- Cybulska, A., & Vossen, P. (2011, June). Historical event extraction from text. In Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (pp. 39-43). Association for Computational Linguistics. link

What do you like in boardgames (P4)

Playing boardgames is a wonderfull hobby that is growing in popularity in the last years. Since 2000, the website BoardGameGeek (BGG) provides a complete database about boardgames and users playing boardgames around the world. Users provide also stats and ratings that evaluate the popularity of each game according to several criteria, including a rating and the number of users voting the game, the community opinion about the playability of the game with respect to the number of players, the community opinion about how much the game is language dependant, the game complexity (called weight) (see for example the stats for the game Gloomhaven). Moreover, BGG provide access to users comments on games that are often associated with a review score given to the game through their BGG API.

Goal of the project is to study the user comments in order to understand in which comments the following **aspects** of boardgaming are mentioned (definitions a taken from the Italian Goblinpedia):

- **luck** or **alea**: all those game elements independent of player intervention, introduced by game mechanics outside the control of the players.
- **bookkeeping**: manual recording of data and potentially automatic or semi-automatic game processes, including also the need of continuosly accessing the rulebook for reference.
- **downtime**: unproductive waiting time between one player turn and the next. By unproductive we mean not only having nothing (or little) to do, but also nothing (or little) to think about.
- **interaction**: the degree of influence that one player's actions have on the actions of the other participants.

- **bash the leader**: when, to prevent the victory of whoever is first, the players are forced to take actions against him, often to the detriment of their own advantage or in any case without gaining anything directly. At the table, the unfortunate situation can arise whereby one or more must "sacrifice" themselves to curb the leader and let the others benefit from this conduct.
- **complicated vs complex**: A game is complicated the more the rules are quantitatively many and qualitatively equipped with exceptions. Once you understand and learn all the variables, a game (that is only) complicated is not difficult to master. In a complicated game, solving a problem leads to immediate, certain and predictable results.
  A game is as complex as the repercussions of one's actions are difficult to predict and master. Even once you understand and learn all the variables, a complex game is still difficult to master. In a complex game, solving one problem leads to other problems.

For each aspect, the project will develop a method to find references to that aspect in user comments.

Datasets

Game metadata as well as users comments and evaluation score may be accessed through thee BGG API.

References

Rajagopal, D., Cambria, E., Olsher, D., & Kwok, K. (2013, May). A graph-based approach to commonsense concept extraction and semantic similarity detection. In *Proceedings of the 22nd International Conference on World Wide Web* (pp. 565-570). link

Rana, T. A., & Cheah, Y. N. (2016). Aspect extraction in sentiment analysis: comparative analysis and survey. Artificial Intelligence Review, 46(4), 459-483. link

Nazir, A., Rao, Y., Wu, L., & Sun, L. (2020). Issues and challenges of aspect-based sentiment analysis: A comprehensive survey. IEEE Transactions on Affective Computing. link

Explain you opinion (P5)

> In collaboration with Emanuele Guidotti, Università della Svizzera Italiana (USI)

Born Classifier is a text classification algorithm inspired by the notion of superposition of states in quantum physics. Born provides good classification performance, explainability, and computational efficiency. In this project, the goal is to exploit the Born explanation in order to use it for Aspect Based Sentiment Analysis. In particular, the main idea to to proceed as follows:

1. Perform a sentiment analysis classification of documents using Born
2. Extract the explanation features for each pair of documents and predicted labels
3. Analyze the explanatory features in order to group them in candidate aspects
4. Associate each aspect to a specific sentence or portion of the text
5. Predict the sentiment for the sentence or text portion using the trained Born classifier
6. Associate then a (potentially different) sentiment to each sentence or text portion according to the aspect

Finally, evaluate the quality of the results for each aspect.

Dataset

Any dataset supporting ABSA. See for example here.

References

- Emanuele Guidotti and Alfio Ferrara. Text Classification with Born's Rule. *Advances in Neural Information Processing Systems*, 2022.
- Schouten, K., & Frasincar, F. (2015). Survey on aspect-level sentiment analysis. IEEE Transactions on Knowledge and Data Engineering, 28(3), 813-830. link
- Rana, T. A., & Cheah, Y. N. (2016). Aspect extraction in sentiment analysis: comparative analysis and survey. Artificial Intelligence Review, 46(4), 459-483. link

No Country for Old Men (P6)

Today the debate about the risks associated with the pervasive use of artificial intelligence in human communication is of great importance. In particular, Large Language Models are at the center of these concerns as the use of these technologies can potentially lead to the spread and increase of disinformation, erroneous beliefs, social bias and the spread of stereotypes that can affect the ethical dimension of communication.

In this framework, the project aims to propose a methodology and techniques to test existing pre-trained models and measure the ethical risk they may entail. In particular, you are asked to:

1. Choose a model and a task (e.g., text classification, question answering, data summarization, text generation, machine translation, word embedding).
2. Precisely define a measurable research question regarding a possible discriminatory behavior of the model (e.g., is there a gender bias in predicting occupation? Is there a stereotype regarding ethnicity with respect to deviant social behavior ?). Students are encouraged to expand the list beyond examples, but clearly define the goal.
3. Define a test method with which to evaluate the behavior of the chosen model.
4. Produce measurable results that allow to estimate the risk associated with the chosen pre-trained model.

Dataset

- Any pre-trained model from https://huggingface.co/ or from any other source

References

- Garg, N., Schiebinger, L., Jurafsky, D., & Zou, J. (2018). Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, *115*(16), E3635-E3644.
- Bolukbasi, T., Chang, K. W., Zou, J. Y., Saligrama, V., & Kalai, A. T. (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in neural information processing systems*, *29*.
- Kiritchenko, S., & Mohammad, S. M. (2018). Examining gender and race bias in two hundred sentiment analysis systems. *arXiv preprint arXiv:1805.04508*.

- Dixon, L., Li, J., Sorensen, J., Thain, N., & Vasserman, L. (2018, December). Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society* (pp. 67-73).

Who you are? (P7)

> In collaboration with Emanuele Guidotti, Università della Svizzera Italiana (USI)

Entity Disambiguation is the task of linking mentions of ambiguous entities to their referent entities in a knowledge base such as Wikipedia. Born Classifier is a text classification algorithm inspired by the notion of superposition of states in quantum physics. Born provides good classification performance, explainability, and computational efficiency. In this project, the goal is to exploit the Born explanation in order to try to perform entity disambiguation. In particular, the main idea to to proceed as follows:

1. Perform Named Entity Recognition as a classification process with Born, where the text is the input and the entities mentioned in the text are the target
2. Extract the explanation features for each pair of documents and predicted entities
3. Analyze the explanatory features in order to group see if ambiguous entities (i.e., entities with the same name) are associated with different explanatory features in different documents
4. Try to use the explanatory features in order to distinguish different classes of equivalence referring to the same ambiguous entity
5. You can also collect different Wikipedia pages potentially corresponding to the same ambiguous entities (e.g., Berlin as the capital city of Germany or Berlin as the TV series) and run the classification on these documents to match the explanations of the original dataset with those of the Wikipedia pages and select the right sense for the entity
6. Finally, evaluate the quality of the results for the disambiguation task.

Dataset

Any dataset supporting Entity Disambiguation. See for example WikilinksNED, AIDA CoNLL-YAGO, or AQUAINT.

References

- Emanuele Guidotti and Alfio Ferrara. Text Classification with Born's Rule. *Advances in Neural Information Processing Systems*, 2022.
- Yotam Eshel, Noam Cohen, Kira Radinsky, Shaul Markovitch, Ikuya Yamada, and Omer Levy. 2017. Named Entity Disambiguation for Noisy Text. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 58–68, Vancouver, Canada. Association for Computational Linguistics.
- Hoffart, J., Yosef, M. A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., ... & Weikum, G. (2011, July). Robust disambiguation of named entities in text. In *Proceedings of the 2011 conference on empirical methods in natural language processing* (pp. 782-792).

Write me (P8)

The extraction of information elements from unstructured text is a widely addressed research problem in the field of text mining. E-mail and chat messages are characterized not only by a partially structured form, which includes - for example - fixed formulas and metadas, but also by a dialogical and dynamic nature. Following these considerations, the research project could approach the problem of extracting information content from e-mail message texts on three different levels, not necessarily all mandatory:

Structural elements: Extract the formal elements of the single e-mail messages, such as the pleasantries and the signature, but also themes and topics.

Non-structural elements: All the elements that are *expected* in an email but are not structural, such as arguments. The problem of argument mining in particular has received and continues to receive great attention in the scientific literature.

Dialogues: Extension of the previous tasks to a dialogical environment, such as a chat.

Dataset

- Enron Email Data
- Hillary Clinton Emails
- Apache Email Archives
- NPS CHat Corpus
- Fraudulent E-mail Corpus

References

- Agrawal, S., Chakrabarti, K., Chaudhuri, S., & Ganti, V. (2008). Scalable ad-hoc entity extraction from text collections. *Proceedings of the VLDB Endowment*, *1*(1), 945-957.
- Al-Moslmi, T., Ocaña, M. G., Opdahl, A. L., & Veres, C. (2020). Named entity extraction for knowledge graphs: A literature overview. *IEEE Access*, *8*, 32862-32881.
- Hong, T., Cho, J., Yu, H., Ko, Y., & Seo, J. (2023). Knowledge-grounded dialogue modelling with dialogue-state tracking, domain tracking, and entity extraction. *Computer Speech & Language*, *78*, 101460.

Find Hidden Entities in Wikipedia Articles (P9)

> Instructor: Sergio Picascia, Department of Computer Science, Università degli Studi di Milano

Wikipedia, the greatest free online encyclopedia, contains millions of articles, connected to each other via hyperlinks. Reading through an entire page in order to retrieve valuable information may result in a daunting and arduous task. Thus, one solution could consist in exploring the entities linked to the page using a tool for knowledge extraction and identifying triples having the following structure: <agent>, <relation>, <target>.

Dataset

- Wikidata5m. Wikidata5m is a million-scale knowledge graph dataset with aligned corpus. This dataset integrates the Wikidata knowledge graph and Wikipedia pages. Each entity in Wikidata5m is described by a corresponding Wikipedia page, which enables the evaluation of link prediction over unseen entities. (link)
- Dataset Card for Wikipedia. The dataset is extracted from a Wikipedia dump and split for different languages. Each example contains the plain text of one Wikipedia article, with markdown and references already removed. (link)

References

- Hachey, B., Radford, W., Nothman, J., Honnibal, M., & Curran, J. R. (2013). Evaluating entity linking with wikipedia. Artificial intelligence, 194, 130-150. (link)
- Mirrezaei, S. I., Martins, B., & Cruz, I. F. (2015). The triplex approach for recognizing semantic relations from noun phrases, appositions, and adjectives. In *The Semantic Web: ESWC 2015 Satellite Events: ESWC 2015 Satellite Events, Portorož, Slovenia, May 31–June 4, 2015, Revised Selected Papers 12* (pp. 230-243). Springer International Publishing.
- Yang, Y., Zhou, S., & Liu, Y. (2023). Bidirectional relation-guided attention network with semantics and knowledge for relational triple extraction. *Expert Systems with Applications*, *224*, 119905.

Projects in collaboration with Volocom technology

Evergreen Article Classification (V1)

An evergreen article is a piece of content that remains relevant and valuable over a long period of time, often addressing timeless topics that are always of interest to readers. The goal of evergreen article classification is to identify articles that are evergreen and distinguish them from those that are time-sensitive or trending. The project aims to develop a model that can accurately classify evergreen articles based on their content.

Dataset

`train.csv`: A full training dataset with the following attributes:

- `url`: Url of the webpage to be classified
- `boilerplate`: Boilerplate text
- `commonLinkRatio_1`: # of links sharing at least 1 word with 1 other links / # of links
- `commonLinkRatio_2`: # of links sharing at least 1 word with 2 other links / # of links
- `commonLinkRatio_3`: # of links sharing at least 1 word with 3 other links / # of links
- `commonLinkRatio_4`: # of links sharing at least 1 word with 4 other links / # of links
- `compression_ratio`: Compression achieved on this page via gzip (measure of redundancy)
- `is_news`: True (1) if StumbleUpon's news classifier determines that this webpage is front-page news
- `lengthyLinkDomain`: True (1) if at least 3 's text contains more than 30 alphanumeric characters
- `linkwordscore`: Percentage of words on the page that are in hyperlink's text

- `news_front_page`: True (1) if StumbleUpon's news classifier determines that this webpage is front-page news
- `non_markup_alphanum_characters`: Page's text's number of alphanumeric characters
- `spelling_errors_ratio`: Ratio of words not found in wiki (considered to be a spelling mistake)
- `label`: User-determined label. Either evergreen (1) or non-evergreen (0); available for train.tsv only

Fake News Detection (V2)

Fake news refers to misinformation or false information presented as news, often with the intent to deceive. The proliferation of fake news can have significant impacts on public opinion and behavior, making its detection a critical task. The goal of fake news detection is to identify and classify news articles as either true or false. This project aims to develop a model that can accurately classify test articles based on their veracity, thereby contributing to the fight against misinformation.

Dataset

`train.csv`: A full training dataset with the following attributes:

- `id`: unique id for a news article
- title: the title of a news article
- `author`: author of the news article
- `text`: the text of the article; could be incomplete
- `label`: a label that marks the article as potentially unreliable

News Broadcast Analysis (V3)

This project focuses on the comprehensive analysis of news broadcast transcripts, targeting three main objectives: event detection, summarization, and the identification of mentions of the political party "Lega." The overarching goal is to develop a robust system capable of dissecting news broadcasts into distinct segments, generating accurate and concise summaries for each segment, and evaluating the presence and context of references to the political party "*Lega*." By achieving these objectives, the project aims to enhance the understanding of broadcast news content, improve media monitoring capabilities, and provide valuable insights into media coverage and political discourse.

The project may focus on one or more of the following tasks:

Event Detection: Develop an algorithm that can automatically identify and segment different news services within broadcast transcripts. The system should be able to discern transitions between different topics, such as shifting from politics to sports or weather, based on linguistic and contextual cues.
Summarization: Implement a summarization technique that can generate concise and coherent summaries of each identified segment.The summaries should capture the essential information and key points of each segment, providing a quick overview for users.
Political Reference Evaluation: Create a method to detect and evaluate mentions of the political party "Lega" within the segments.

Dataset

News Broadcast Transcripts Dataset. The dataset contains transcripts of news broadcasts from various Italian TV channels. Each transcript includes multiple segments covering different news topics.

---

1. Sprugnoli, R., & Tonelli, S. (2019). Novel event detection and classification for historical texts. *Computational Linguistics*, *45*(2), 229-265. liink ↩