Team: **Neural fusion**

Members

1. Sulayman Adewale
2. Nicolas Giot
3. Fisayo Jassey Jabarr
4. Thien Nam Nguyen
5. Kamilla Tulepbergenova

Professor Patricia McManus

ITAI 2376: Deep Learning in Artificial Intelligence -

May 1, 2024

**Comprehensive Report –** Building an Autonomous Agent Creation

**Integration of AI Agent Design:**

Our team embarked on the challenge of designing an AI agent for the "Keyword Hunt" project, leveraging a combination of TF-IDF and Word2Vec techniques. Initially, we grappled with understanding the core concepts behind these methods, particularly their application in natural language processing. Through our meetings, we gained clarity on how TF-IDF captures the significance of keywords within documents and how Word2Vec extracts semantic relationships between words.

With some improved understanding, we proceeded to implement the AI agent, starting with the extraction of keywords using TF-IDF. This involved utilizing the TfidfVectorizer from the: sklearn.feature\_extraction.text module to transform text data into TF-IDF features. However, we encountered challenges in fine-tuning the vectorizer parameters to achieve optimal keyword extraction. With guidance from our team members and further exploration of the documentation, we refined our approach and successfully generated TF-IDF keywords from the provided clues.

Next, we explored Word2Vec, a method renowned for capturing semantic similarities among words. Utilizing Word2Vec involved tokenizing the clues and training a Word2Vec model using genism. models.Word2Vec class. However, selecting optimal parameters, such as vector size and window size, presented challenges. Despite conducting additional research and experimenting with various GPTs (Generative Pre-trained Transformers), we encountered difficulties in determining the most suitable parameters. Consequently, we iteratively adjusted these parameters to improve the model's performance.

**Documentation:**

As we progressed with the implementation, we documented our code changes and additions to maintain clarity and facilitate knowledge sharing within the team. This documentation included detailed explanations of the TF-IDF and Word2Vec processes, along with insights into the challenges encountered and the decision-making process behind parameter selection.

Our documentation served as a valuable reference, enabling team members to understand the rationale behind specific code decisions and providing a framework for future modifications. Additionally, we utilized inline comments and descriptive variable names to enhance readability and comprehension, ensuring that our code was accessible to all team members.

Team Collaboration:

Effective collaboration was paramount to the success of our project. We organized regular team meetings to discuss progress, address challenges, and allocate tasks based on individual strengths and interests. Each team member actively contributed to different aspects of the project, sharing insights, troubleshooting issues, and offering support where needed.

Roles and responsibilities were clearly defined, with one team member focusing on TF-IDF implementation, another on Word2Vec, and others on documentation and code review. This division of labor facilitated efficient task execution and ensured that all project components were thoroughly addressed.

Reflection:

Reflecting on our journey, we acknowledge the initial apprehension and uncertainty we faced when tackling complex AI techniques for the first time. However, through perseverance and collaboration, we overcame these challenges and gained confidence in our abilities to implement advanced algorithms for text processing.

Our experience with the "Keyword Hunt" project has been invaluable, providing us with practical insights into AI and NLP concepts and their real-world applications. Moving forward, we are excited to further explore these technologies and continue our journey of learning and growth in the field of artificial intelligence.

Going forward

We are recommend the following steps to further enhance our AI agent and continue our learning journey in artificial intelligence:

* Explore Advanced NLP Techniques: Delve deeper into advanced natural language processing (NLP) techniques beyond TF-IDF and Word2Vec. This could include methods such as BERT, GPT, and Transformer-based models, which have demonstrated state-of-the-art performance in various NLP tasks.
* Experiment with Pre-trained Models: Utilize pre-trained models available through libraries like Hugging Face Transformers to leverage powerful NLP capabilities without the need for extensive training. Experiment with fine-tuning these models on domain-specific data to tailor them to our specific application requirements.
* Enhance Keyword Extraction: Refine and expand the keyword extraction capabilities of our AI agent by incorporating additional techniques such as graph-based algorithms, keyword clustering, and topic modeling. This will enable more comprehensive and accurate identification of relevant keywords within text data.
* Implement Interactive Features: Introduce interactive features to the AI agent, allowing users to provide feedback and refine keyword extraction results in real-time. This could involve incorporating user preferences, adjusting parameters dynamically, and providing visualization tools to aid in result interpretation.
* Try to apply AI in Real-world Scenarios: Seek opportunities to apply AI techniques in real-world scenarios beyond the Keyword Hunt project. Explore interdisciplinary applications in fields such as healthcare, finance, education, and sustainability, where AI can make meaningful contributions to solving complex problems and driving innovation.

The most crucial insights from our labs, class and report are:

1. Effective Techniques: TF-IDF and Word2Vec are effective for keyword extraction, but exploring advanced models like BERT can enhance accuracy.
2. Pattern Detection: Regular expressions are valuable for detecting domain-specific terms in clues, improving keyword identification.
3. Collaborative Teamwork: Collaboration facilitates task division and code integration, enhancing project efficiency.
4. Challenges Drive Learning: Overcoming challenges like model selection and error handling provides valuable learning experiences.
5. Practical Applications: The AI agent has practical applications in content analysis, information retrieval, and recommendation systems.

Thank you.

Team: **Neural fusion**

Refences:

1. <https://awsacademy.instructure.com/>
2. Utilize pre-trained models (Hugging Face Transformers: <https://huggingface.co/docs/transformers/en/index>)
3. **TF-IDF & Word2Vec:** Effective for keyword extraction but consider exploring advanced models like BERT for increased accuracy. (<https://scikit-learn.org/>, <https://github.com/piskvorky/gensim/blob/develop/gensim/models/word2vec.py>)

Take 2 :

**Building the AI Agent: A Journey of Learning and Collaboration**

Our team set out to design an AI agent for the "Keyword Hunt" project, employing a combination of TF-IDF and Word2Vec techniques. Initially, these methods were unfamiliar territory. Through dedicated team meetings, we delved deeper into their core concepts, specifically how TF-IDF quantifies keyword importance within documents and how Word2Vec captures semantic relationships between words.

We then embarked on the implementation, starting with TF-IDF keyword extraction. Utilizing scikit-learn's TfidfVectorizer, we transformed text data into TF-IDF features. However, fine-tuning the vectorizer parameters for optimal keyword extraction proved challenging. By leveraging team expertise and documentation resources, we refined our approach and successfully generated relevant TF-IDF keywords from the clues.

Next, we ventured into Word2Vec, renowned for its ability to capture word similarities. We tokenized the clues and trained a Word2Vec model using gensim's Word2Vec class. However, selecting optimal parameters like vector size and window size presented hurdles. While we experimented with various GPTs (Generative Pre-trained Transformers) to explore potential improvements, the challenge of parameter selection persisted. Through iterative adjustments, we ultimately achieved satisfactory performance with Word2Vec.