UNIVERSIDADE
FEDERAL RURAL
DE PERNAMBUCO

Matheus Rodrigues de Souza Félix

Uma Proposta para Agrupamento Automático de Horas de Trabalho

Recife

Junho de 2022

Matheus Rodrigues de Souza Félix

Uma Proposta para Agrupamento Automático de Horas de Trabalho

Artigo apresentado ao Curso de Bacharelado em Sistemas de Informação da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação.

Universidade Federal Rural de Pernambuco – UFRPE
Departamento de Estatística e Informática
Curso de Bacharelado em Sistemas de Informação

Orientadores: Cleviton V. F. Monteiro e Rinaldo Lima

Recife
Junho de 2022

Dados Internacionais de Catalogação na Publicação
Universidade Federal Rural de Pernambuco
Sistema Integrado de Bibliotecas
Gerada automaticamente, mediante os dados fornecidos pelo(a) autor(a)

F316p Félix, Matheus
Uma Proposta para Agrupamento Automático de Horas de Trabalho / Matheus Félix. - 2022.
11 f. : il.

Orientador: Cleviton V. F. Monteiro.
Coorientador: Rinaldo Lima.
Inclui referências e apêndice(s).

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal Rural de Pernambuco, Bacharelado em Sistemas da Informação, Recife, 2022.

1. Agrupamento. 2. Horas de Trabalho. 3. Inteligência Artificial. I. Monteiro, Cleviton V. F., orient. II. Lima, Rinaldo, coorient. III. Título

CDD 004

Matheus Rodrigues de Souza Félix

Uma Proposta para Agrupamento Automático de Horas de Trabalho

Artigo apresentado ao Curso de Bacharelado em Sistemas de Informação da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Bacharel em Sistemas de Informação.

Aprovada em: 03 de Junho de 2022.

BANCA EXAMINADORA

Cleviton V. F. Monteiro
Departamento de Estatística e Informática
Universidade Federal Rural de Pernambuco

Rinaldo Lima
Departamento de Computação
Universidade Federal Rural de Pernambuco

Uma Abordagem para Agrupamento Automático de Horas de Trabalho

Matheus Rodrigues de Souza Félix [1], Cleviton V. F. Monteiro/Rinaldo Lima [1]

[1]Departamento de Estatística e Informática – Universidade Federal Rural de Pernambuco
Rua Dom Manuel de Medeiros, s/n, - CEP: 52171-900 – Recife – PE – Brasil

Resumo. *O registro de horas utilizadas em projetos é uma tarefa recorrente no dia-a-dia de grande parte dos profissionais. Esta tarefa é crucial em processos relacionados a administração e recursos humanos para análise de alinhamento com cronogramas e produtividade. Entretanto, o preenchimento correto e em prazo são pontos importantes para que o ciclo de realização de atividades e registro seja efetivo. Quando o profissional trabalha em diversos projetos de forma alternada no seu cotidiano, o registro dessas atividades tende a ganhar imprecisão. Neste artigo, será apresentada uma proposta para possibilitar a automatização do registro de horas através do uso de técnicas de mineração de texto. O objetivo deste projeto é criar uma abordagem que auxilie o usuário reduzindo horas diárias registrando atividades.*

Abstract. *The recording of hours used in projects is a recurring task in the day-to-day of most professionals. This task is crucial in processes related to administration and human resources for analysis of alignment with schedules and productivity. However, filling in correctly and on time are important points for the cycle of activities and registration to be effective. When the professional works on several projects in an alternating way in his daily life, the record of these activities tends to gain inaccuracy. In this article, a proposal will be presented to enable the automation of time recording through the use of text mining techniques. The objective of this project is to create a facilitator that helps the user by minimizing the daily hours creating records of work performed and increasing the accuracy of the records.*

1. Introdução

Como uma tarefa bem definida no meio organizacional, a gestão do tempo de trabalho referente as horas utilizadas em projetos ou atividades é fundamental para diversas métricas que viabilizam a visão de negócio. Com colaradores regularmente registrando suas horas produtivas, é possível precisar decisões como reajuste de escopo de projeto, realocação de equipe, elaboração de documentos de mapeamento do projeto, entre outras [Senior 2021].

Do lado do colaborador, seu registro de horas também é um indicador importante sobre eventuais dificuldades, isto é, se o colaborador tem uma habilidade avançada em uma determinada área e alguma atividade acaba tomando mais tempo que o esperado, talvez seja indicado a adição de mais um colaborador nesta atividade. Da mesma forma, um colaborador encarado como menos experiente executando uma tarefa entendida como mais difícil, pode ser reajustado em sua posição.

Apesar dos benefícios para os envolvidos, a atividade de registro de horas trabalhadas demanda tempo diário do colaborador. Como essas atividades são recorrentes e manuais, alguns softwares utilizam técnicas para minimizar o esforço como registro de capturas de telas em tempos determinados para , no final do dia, o colaborador lembrar oque estava fazendo e associar a imagem a uma tarefa. Ainda assim, muitas dessas abordagens podem ser consideradas intrusivas quando conectadas a plataformas com envio de dados automatizados.

Uma limitação envolvida no uso de algoritmos tradicionais de aprendizado de máquina é que a relação léxico-temporal presente entre as atividades e seus projetos acaba não sendo abstraída por modelagens tradicionais de dados. Por conta disso, este projeto utiliza heurísticas que contemplam o entendimento desta relação viabilizando uma abordagem mais robusta ao problema.

Neste artigo, será apresentada uma abordagem para agrupar registro de horas em projetos afim de auxiliar o colaborador nas atividades de apontamento de horas. A saída da abordagem apresentada permite o colaborador realizar ajustes conforme desejar. Com o uso de técnicas de mineração de texto em registros de softwares executados, é proposta uma abordagem de agrupamento considerando o nome e hora de execução do processo.

2. Trabalhos Relacionados

O trabalho Framework de captação e categorização automática de registro de horas de trabalho [Junior and Monteiro 2019] visa categorizar horas reportadas a partir de uma abordagem supervisionada que utiliza e utiliza técnicas de aprendizado de máquina em registros de atividades, horas e as categorias de projetos, para realizar a classificação de atividades. Diferente do trabalho de captação e categorização de horas, neste presente trabalho é definida uma abordagem para agrupar as atividades a partir dos seus projetos, sendo assim uma etapa anterior ao trabalho de categorização de horas.

Apesar de ser um problema recorrente no cotidiano de diversos profissionais, não foram encontrados trabalhos relacionados a agrupamento de atividades. Apesar de algumas propostas parecerem similares, não há ferramentas com o objetivo de agrupar as atividades automaticamente para gerar anotações dos seus respectivos projetos.

No aspecto da problemática apresentada, este artigo oferece uma abordagem moldada sobre a perspectiva de privacidade do usuário. Portanto, é importante ressaltar que modelos que possam usufruir de integrações com ambientes externos para coletar e enviar dados de forma automática não estão alinhados com o propósito das abordagens propostas.

3. Materiais e Métodos

A Figura 1 apresenta o fluxo geral do projeto. Como etapa inicial, executada a coleta de dados no ManicTime, onde o usuário poderá extrair o Registro de Atividades e criar sua Relação entre projetos e palavras-chave. Com essas duas informações, é possível realizar a entrada de dados com dois fluxos distintos, sendo estes o de treinar o modelo buscando parâmetros ótimos ou o de utilizar parâmetros definidos por padrão no modelo. Para que ocorra o fluxo de treino, é necessário que o usuário crie uma coluna, na tabela

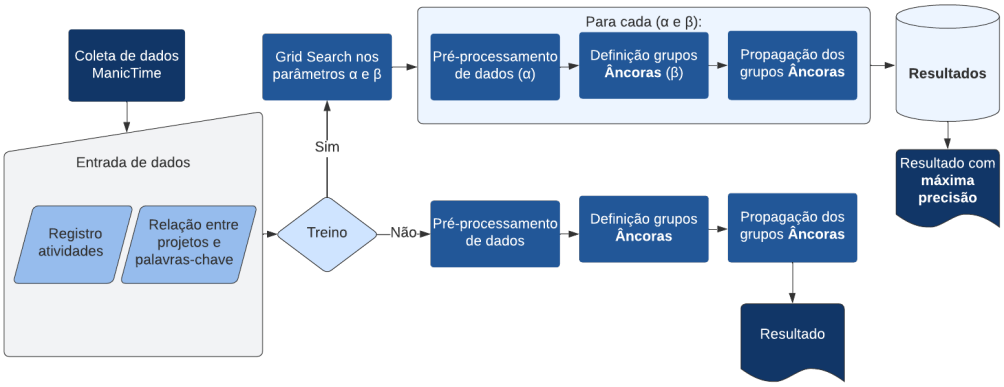

Figura 1. Fluxo geral

gerada pelo ManicTime, identificando o projeto associado a uma atividade. Esta etapa permite que seja executado uma busca por melhores parâmetros específicos para a base do usuário. No caso de não execução da etapa de treinamento, o modelo será executado com parâmetros padrões do projeto, que não garantem retorno ótimo. Nos dois fluxos, ocorrem as etapas de pré-processamento, definição de grupos âncoras e propagação dos grupos âncoras, sendo a etapa de Treino geradora de uma busca de valores para β e α com armazenamento em uma base de dados que retorna os valores que obtiveram maior precisão.

O conjunto de dados utilizado nesse projeto foi coletada a partir do software de rastreio de atividades ManicTime[Time 2008]. Cada software instalado, na máquina do usuário, executa registros de suas atividades no sistema operacional tornando possível, através da aplicação ManicTime, coletar as atividades que estão em execução principal. Cada atividade é coletada e é descrita por um nome, data de início, data de fim, duração e nome do software executor.

Os dados deste trabalho foram coletados de engenheiros de software de uma empresa que colaborou com o projeto. Com participação de três colaboradores registrando três dias de atividades, foi possível coletar 1601 atividades, em média, por engenheiro. Vale ressaltar que cada atividade é analisada individualizada por usuário e os dados de um usuário não são utilizados para gerar informações de projetos de outros usuários. Os engenheiros também incrementaram na tabela de dados coletada a coluna *Projeto*, indicando o nome do projeto associado a atividade para fins de análise de resultado. Além disso, os engenheiros forneceram uma base que relaciona nome do projeto e palavras-chave associadas ao projeto. Na Tabela 1 é possível observar uma amostra de dados ordenados cronologicamente com remoção das colunas data de início e data de fim.

Nome	Duração	Processo	Projeto
dashb - Google Chrome	00:00:05	Google Chrome	Projeto 3
cash - Microsoft Azure	00:00:07	Google Chrome	Projeto 5
MSQL Server Management Studio	00:00:21	Microsoft SQL Server	Projeto 5

Tabela 1. Amostra base de dados Engenheiro 1.

A Tabela 2 apresenta a contagem de atividades nas bases em relação aos projetos encontrados nas bases e os respectivos engenheiros. Neste trabalho foi observado a

existência de classes comuns a todos os Engenheiros como a classe Projeto 6 e Projeto 7. A classe Projeto 6 representa as atividades que os Engenheiros consideraram não produtivas como uso de mídias sociais, execução de softwares de música e etc. Projeto 7 representa uma classe de pesquisa, projeto que conta com relatório de todos os Engenheiros. As demais classes representam atividades individuais de cada Engenheiro que, ocasionalmente , acabaram registrando as mesmas classes. Além disso, podemos observar que existem algumas classes raras de projeto por análise de contagem. Por conta da heurística utilizada neste projeto, estas classes não causam impacto direto nos resultados. Entretanto, em cenário de utilização do projeto, estas classes podem ser pontuais o suficiente para que as entradas de palavras-chave do usuário não tenham uma alta relação léxica com as atividades da classe rara.

Projeto	Engenheiro 1	Engenheiro 2	Engenheiro 3
Projeto 1	-	1	-
Projeto 2	-	1	9
Projeto 3	1168	-	-
Projeto 4	1	-	-
Projeto 5	286	-	-
Projeto 6	107	186	1
Projeto 7	23	2	44
Projeto 8	-	138	-
Projeto 9	-	-	1416
Projeto 10	-	-	234
Total	1703	1361	1740

Tabela 2. Relação de contagem entre projetos e engenheiros.

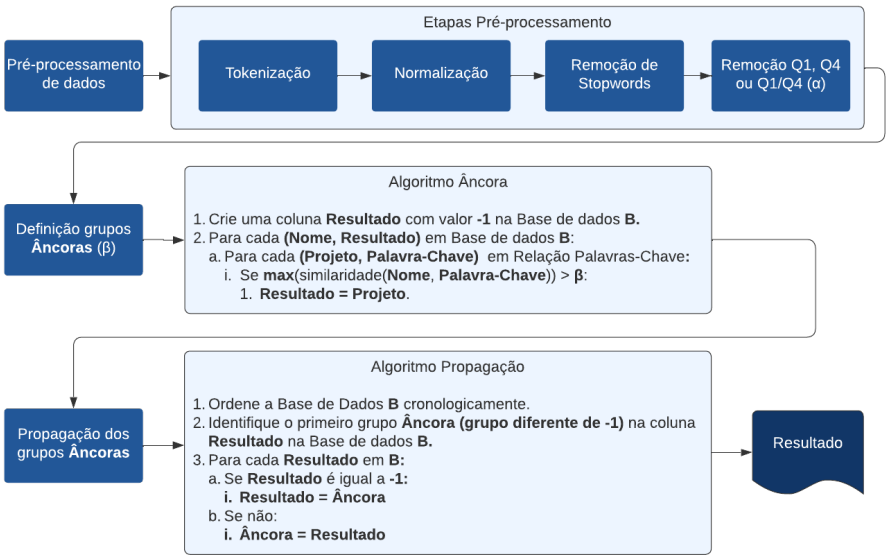

Figura 2. Fluxo detalhado

De uma visão mais interna dos fluxos, é possível observar na Figura 2 um maior detalhe nas etapas envolvidas em cada um dos blocos do fluxo geral que contam com

etapas internas. A etapa de pré-processamento de dados realizar quatro etapas internas responsáveis por criar a visão de dados das etapas posteriores. A tokenização é o processo de separar uma sentença em palavras. Normalização de texto consiste na remoção de caracteres especiais e transformação de letras em minúsculas, garantindo que palavras escritas de forma diferente mas com significados iguais apresentem maior similaridade léxica. Na remoção de **stopwords**, palavras de baixo valor discriminatório para identificar um projeto (preposições, artigos e etc), foram removidas palavras em português e inglês contidas na base de dados de **stopwords** do Natural Language Tool Kit (NLTK). A última etapa consiste numa heurística de melhoria da base de dados utilizando a técnica de remoção de quartil. Todas as etapas, exceto a última, também são realizadas no conjunto de palavras-chave do usuário. [Bird 2006] [Mehanna and Mahmuddin 2021]

Para gerar visões de bases de dados com capacidade de remover palavras que pudessem gerar ruídos, a técnica de remoção de quartil foi utilizada com visões de remoção do primeiro quartil(**Q1**), quarto quartil(**Q4**) e de ambos os quartis(**Q1/Q4**). O primeiro quartil contêm palavras menos frequentes e o quarto as palavras mais frequentes. A heurística por trás dessa decisão consiste em remover palavras que apareçam pontualmente como a execução de alguma música ao longo do dia ou palavras de muita frequência como um nome de software escrito junto ao nome da tarefa. Quando realizada a identificação das palavras a serem removidas, todo token da coluna de nomes com similaridade maior que α em relação as palavras removidas por quartis é removido da base. Como função de similaridade, foi utilizada a função *ratio* da ferramenta FuzzyWuzzy[Geek 2010] que implementa uma variação do algoritmo de Levenshtein [ZHAO et al. 2009] com ponderação da diferença da quantidade de caracteres entre as palavras de entrada para transformação do resultado em uma taxa. Com isso, cada base foi replicada da seguinte forma: primeiro quartil removido, quarto quartil removido e ambos quartis removidos.[H P et al. 2018]

A etapa seguinte, apresentada na Figura 2 como *Definição grupos Âncoras*, consiste em executar um cálculo de similaridade léxico na coluna de nomes de tarefas filtrados com as palavras-chave dadas como entrada pelo usuário. Este passo é o gerador de *âncoras* para grupos, isto é, caso o limiar de similaridade entre alguma palavra da tarefa e palavra-chave do usuário seja maior que β, esta tarefa é considerada pertencente ao grupo relacionado a palavra-chave. Nesta etapa, é possível expandir o vocabulário relacionado de palavras do usuário para uso posterior. O resultado desta etapa é armazenado numa nova coluna chamada *Resultado*, inicialmente tendo todos os valores definidos para -1. Caso a similaridade máxima encontrada na busca entre cada nome da tabela Registro de Atividades do usuário e palavras-chaves tenha um valor superior a β, o valor de Resultado para a linha do Registro de Atividades deixa de ser -1 e se torna valor Projeto associado a palavra-chave. Nesta etapa, alguns elementos já são associados a seus grupos, por similaridade léxica, enquanto outros continuam com a identificação -1 para atribuição posterior, responsável pela análise de atividades próximas temporalmente que poderão adquirir o mesmo rótulo.

As âncoras geradas na etapa anterior são utilizadas no método de propagação baseada em proximidade temporal. Este algoritmo consiste em ordenar a base cronologicamente, atribuir como âncora inicial o primeiro grupo da coluna resultado que aparecer na base e, para cada elemento da coluna resultado na base, se o valor for -1, substituir o valor

pela âncora, caso contrário, âncora recebe o valor de resultado. Esta etapa faz com que atividades sem atribuição de grupos recebam seus rótulos por estarem em proximidade temporam com alguma atividade âncora. Na heurística utilizada, ao percorrer a coluna resultado e encontrar um valor diferente de -1, isto é, uma nova âncora, a âncora definida para atualização de valores nulos é substituída pela mais recente.

Nesta abordagem, é possível contemplar duas relações. Tarefas que apareçam distante temporalmente, mas tenham proximidade léxica, estarão, provavelmente, associadas a uma mesma âncora. Tarefas que tenham baixa similaridade léxica com seu grupo mas estejam próximas temporalmente serão adicionadas aos seus grupos na etapa de propagação de grupos. A abordagem utilizada neste artigo presume que estas relações distintas e implícitas condicionam a relação de pertencimento entre as atividades e seus projetos.

4. Resultados (Estudo de Caso)

A técnica de remoção de quartil foi fundamental para a normalização do vocabulário e melhoria de resultados. Entretanto, este passo é heurístico e apresentou resultados consistentes para o domínio de dados presentes neste trabalho. Como as diferentes bases tem relação no domínio de dados, é esperado que para estas bases o resultado ótimo de remoção seja o mesmo, requisitando a etapa de treino para buscar o modelo ideal para novos domínios.

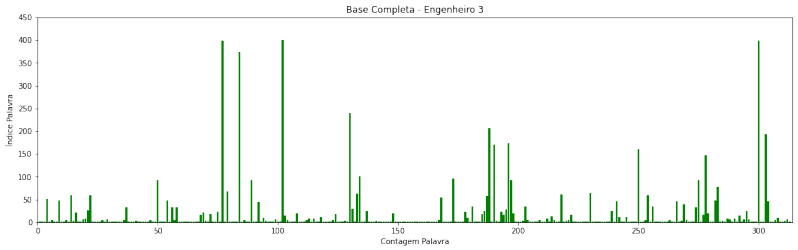

Figura 3. Contagem de palavras base de dados Engenheiro 3

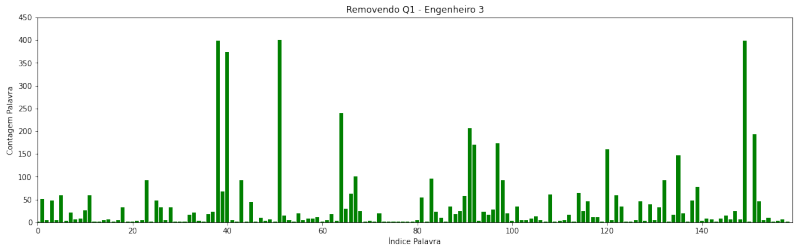

Figura 4. Contagem de palavras base de dados Engenheiro 3 removendo primeiro quartil

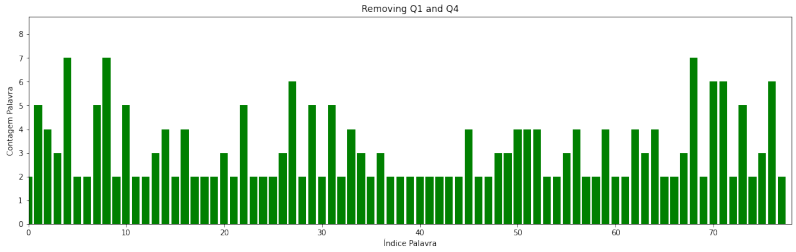

Figura 5. Contagem de palavras base de dados Engenheiro 3 removendo primeiro e quarto quartil

Nas Figuras 3, 4 e 5 é possível observar a diferença que ocorre nos vocabulários durante a remoção dos quartis. A remoção do primeiro quartil remove o trecho com palavras de baixa aparição, provavelmente associadas a execuções pontuais como acessos a arquivos de música ou mídias digitais. Entretanto, com a remoção do primeiro e do quarto quartil, todas as palavras se tornam comuns, promovendo uma baixa taxa de discrimição.

A proposta com remoção de quartil promoveu resultados consistentes, exceto na base do Engenheiro 1, que por contar com muitas atividades com nomes relacionados a logs de mais baixo nível acabou tornando as palavras-chaves de entrada do usuário pouco relevantes para criação de âncoras. Foi realizado uma busca completa de parâmetros nas taxas de similaridade β e α entre o intervalo de 10 a 100 com salto de 10, entre os três modelos de base com remoção de quartil, para cada base de dados.

Base de dados	Base de Dados	Precisão	β	α
Engenheiro 1	Sem Primeiro Quartil	4,92%	80	10
Engenheiro 2	Sem Primeiro Quartil	74,84%	60	10
Engenheiro 3	Sem Primeiro Quartil	84,65%	50	10

Tabela 3. Tabela de resultados da busca de parâmetros em todas as bases.

Base de dados	Base de Dados	Precisão	β	α
Engenheiro 1	Base Completa	3,45%	80	10
Engenheiro 2	Base Completa	71,56%	60	10
Engenheiro 3	Base Completa	79,95%	50	10

Tabela 4. Tabela de resultados da busca de parâmetros em base completa.

A Tabela 3 demonstra o resultado encontrado após a busca de parâmetros em todas as visões das bases de dados. A coluna *Base de Dados* indica qual modelagem da remoção de quartil utilizada. Na Tabela 4, como comparação, é possível analisar a redução da precisão utilizando a busca de parâmetros apenas na base completa, isto é, sem filtro de quartil. É importante salientar que apesar dos resultados serem independentes, a busca de parâmetros entre as bases foi consistente em relação ao Engenheiro 2 e 3, podendo utilizar este parâmetro base para novas avaliações. Como teste dos parâmetros encontrados, foram utilizados os parâmetros médios entre as bases Engenheiro 2 e Engenheiro 3 numa partição de bases separadas da base de busca, trazendo uma degradação de precisão de aproximadamente 5% para cada base. Como as bases tem um domínio de projetos e atividades muito similares a degradação alterando os parâmetros ótimos tende a ser reduzida.

5. Discussão e Trabalhos futuros

Pode-se observar que a relação léxico-temporal identificada neste trabalho e abordada através de heurísticas foi capaz de obter resultados satisfatórios que podem minimizar o esforço do usuário na relação de atividades aos seus projetos. Entretanto, a abstração desta relação pode ser melhor compreendida com análises de bases mais extensas para buscar modelagens com um poder maior de abstração para esta relação e melhorar os resultados.

É importante observar que apesar da consistência nos parâmetros, eles estão sendo analisados em um mesmo domínio, podendo sofrer variações em outras bases. Além disso, existe uma alta dependência da entrada do usuário e da etapa de treinamento para localização do parâmetro ótimo. A etapa de treinamento também pode contar com maior granuralização dos parâmetros β e α. Também como incremento da busca de parâmetros ótimos, adicionar camadas de teste para novas sequências de pré-processamento, com possibilidade de identificação e tratamento de logs ou diretórios. Outra possibilidade é o reajuste da heurística de propagação, podendo haver avaliação de heurísticas mais robustas de propagação de label.

Como melhoria contínua, é possível realizar a expansão do vocabulário de palavras-chave adicionando na relação de entrada do usuário as palavras que forem identificadas como pertencentes a um projeto mas não pertencem a relação de palavras-chave dadas pelo usuário como entrada. Este processo pode ser realizado a medida que o usuário solicita novas respostas ao modelo.

Uma maior base de dados permite analisar abordagens de melhoria com utilização de embeddings[Dridi et al. 2022] para vetorização das atividades com abstração semântica através do tempo. Esta abordagem requer algum grau de alteração nas abordagens tradicionais de embeddings, dado que o objetivo principal é entender uma relação semântica que, neste trabalho, se dá através do tempo. Apesar de não haver publicações similares a esta proposta, é tangível o ajuste dos vetores através de um fator entre a aparição de uma palavra em uma respectiva faixa de tempo, sendo necessário realizar buscas para definir valores ótimos de faixa assim como da relação que irá gerar esta associação léxico-temporal.

Referências

Bird, S. (2006). NLTK: the natural language toolkit. In Calzolari, N., Cardie, C., and Isabelle, P., editors, *ACL 2006, 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, Sydney, Australia, 17-21 July 2006*. The Association for Computer Linguistics.

Dridi, A., Gaber, M. M., Azad, R. M. A., and Bhogal, J. (2022). Vec2dynamics: A temporal word embedding approach to exploring the dynamics of scientific keywords - machine learning as a case study. *Big Data Cogn. Comput.*, 6(1):21.

Geek, S. (2010). Seatgeek/fuzzywuzzy: Fuzzy string matching in python. https://github.com/seatgeek/fuzzywuzzy. Accessed: 2022-06-02.

H P, V., Poornima, B., and Sagar, B. (2018). *Detection of Outliers Using Interquartile Range Technique from Intrusion Dataset*, pages 511–518.

Junior, E. J. S. and Monteiro, . C. V. F. (2019). Framework de captação e categorização automática de registro de horas de trabalho. *PIBIC/CNPq/UFRPE*.

Mehanna, Y. S. and Mahmuddin, M. (2021). The effect of pre-processing techniques on the accuracy of sentiment analysis using bag-of-concepts text representation. *SN Comput. Sci.*, 2(4):237.

Senior (2021). Gestão de ponto eletrônico: O que É, como fazer e principais vantagens. https://www.senior.com.br/blog/rh-tudo-sobre-gestao-ponto-eletronico. Acessed: 2022-06-03.

Time, M. (2008). Do you ever have problems like these? https://www.manictime.com/. Acessed: 2022-06-03.

ZHAO, Z.-p., YIN, Z.-m., WANG, Q.-p., XU, X.-z., and Jiang, H.-F. (2009). An improved algorithm of levenshtein distance and its application in data processing: An improved algorithm of levenshtein distance and its application in data processing. *Journal of Computer Applications*, 29.